

Sidnei Renato Silveira
Adriana Sadowski de Souza
Antônio Rodrigo Delepiane de Vit
Fábio José Parreira
Guilherme Bernardino da Cunha
Nara Martini Bigolin
Vinicius Gadis Ribeiro



Experiências de Desenvolvimento de Sistemas de Informação Inteligentes

Sidnei Renato Silveira
Adriana Sadowski de Souza
Antônio Rodrigo Delepiane de Vit
Fábio José Parreira
Guilherme Bernardino da Cunha
Nara Martini Bigolin
Vinicius Gadis Ribeiro



Experiências de Desenvolvimento de Sistemas de Informação Inteligentes

Editor Chefe

Dr Washington Moreira Cavalcanti

Autores

Sidnei Renato Silveira
Adriana Sadowski de Souza
Antônio Rodrigo Delepiane de Vit
Fábio José Parreira
Guilherme Bernardino da Cunha
Nara Martini Bigolin
Vinicius Gadis Ribeiro

Conselho Editorial

Dr. Lais Brito Cangussu
Dr. Rômulo Maziero
Msc Jorge dos Santos Mariano
Dr Jean Canestri
Msc Daniela Aparecida de Faria
Dr Paulo Henrique Nogueira da Fonseca
Dr Marcos Pereira dos Santos
Msc Edgard Gonçalves da Costa

Projeto Gráfico e Diagramação

Departamento de arte Synapse Editora

Editoria de Arte

Maria Aparecida Fernandes

Revisão

Os Autores

2025 by Synapse Editora

Copyright © Synapse Editora

Copyright do Texto © 2025 Os autores

Copyright da Edição © 2025 Synapse Editora

Direitos para esta edição cedidos à

Synapse Editora pelos autores.

Todo o texto bem como seus elementos, metodologia, dados apurados e a correção são de inteira responsabilidade dos autores. Estes textos não representam de forma alusiva ou efetiva a posição oficial da Synapse Editora.

A Synapse Editora não se responsabiliza por eventuais mudanças ocorridas nos endereços convencionais ou eletrônicos citados nesta obra.

Os livros editados pela Synapse Editora, por serem de acesso livre, *Open Access*, é autorizado o download da obra, bem como o seu compartilhamento, respeitando que sejam referenciados os créditos autorais. Não é permitido que a obra seja alterada de nenhuma forma ou usada para fins comerciais.

O Conselho Editorial e pareceristas convidados analisaram previamente todos os manuscritos que foram submetidos à avaliação pelos autores, tendo sido aprovados para a publicação.



Compartilhando conhecimento

2025

S569e Silveira, Sidnei Renato

Experiências no Desenvolvimento de Sistemas de Informação Inteligentes.
Autores: Sidnei Renato Silveira, Adriana Sadowski de Souza,
Antônio Rodrigo Delepiane de Vit, Fábio José Parreira, Guilherme Bernardino da Cunha,
Nara Martini Bigolin, Vinicius Gadis Ribeiro.
Belo Horizonte, MG: Synapse Editora, 2025, 108 p.

Formato: PDF

Modo de acesso: World Wide Web

Inclui bibliografia

ISBN: 978-65-88890-54-7

DOI: 10.63951/synapse978-65-88890-54-7

1. Desenvolvimento 2. Sistemas de Informação, 3. Sistemas Inteligentes,
4. Metodologias Ágeis, 5. Desenvolvimento Computacional.

I. Experiências no Desenvolvimento de Sistemas de Informação Inteligentes.

II. Sidnei Renato Silveira, Adriana Sadowski de Souza,
Antônio Rodrigo Delepiane de Vit, Fábio José Parreira,
Guilherme Bernardino da Cunha, Nara Martini Bigolin, Vinicius Gadis Ribeiro.

CDD: 600

CDU: 60 - 681.3

SYNAPSE EDITORA

Belo Horizonte – Minas Gerais

CNPJ: 20.874.438/0001-06

Tel: + 55 31 98264-1586

www.editorasynapse.org

editorasynapse@gmail.com



Compartilhando conhecimento

2025

PREFÁCIO

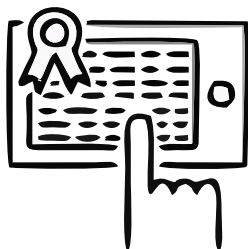
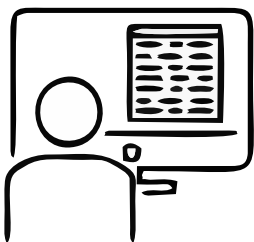
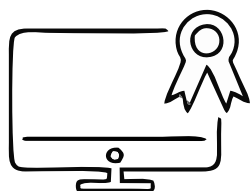
Este e-book apresenta uma série de experiências, compreendendo a área de IA (Inteligência Artificial), desenvolvidas pelo grupo de pesquisa IATE (Inteligência Artificial e Tecnologia Educacional) da UFSM (Universidade Federal de Santa Maria) – Campus Frederico Westphalen/RS. O grupo de pesquisa tem se dedicado ao desenvolvimento de Sistemas de Informação Inteligentes, aplicando diferentes técnicas de IA, tais como Sistemas Especialistas, Sistemas de Recomendação, Sistemas Hipermídia Adaptativos, Raciocínio Baseado em Casos e Redes Neurais Artificiais, entre outras. Alguns dos resultados dos projetos desenvolvidos foram apresentados em eventos da área de Computação, além de serem publicados em anais e periódicos científicos. As experiências apresentadas neste e-book são fruto do trabalho do grupo de pesquisa, desenvolvido no período compreendido entre os anos de 2014 a 2024.

Atualmente, com o avanço da IA, várias áreas estão estudando os impactos da aplicação desta tecnologia, destacando desde os aspectos tecnológicos até os que envolvem as questões éticas. Esperamos que este e-book auxilie pesquisadores da área de Sistemas de Informação, que tenham interesse no desenvolvimento de aplicações utilizando IA.

Os Autores

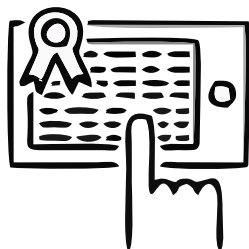
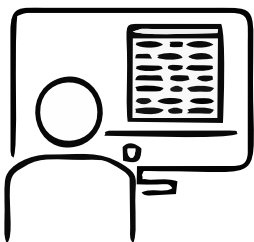
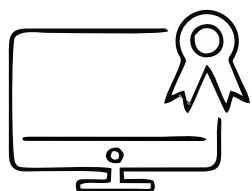


Compartilhando conhecimento
2025



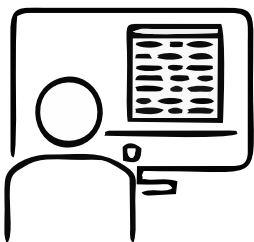
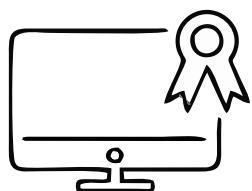
SUMÁRIO

1 INTELIGÊNCIA ARTIFICIAL E O DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO INTELIGENTES	8
1.1 Aplicação de Técnicas de IA no Desenvolvimento de Sistemas de Informação Inteligentes	10
2 EXPERIÊNCIAS NO DESENVOLVIMENTO DE SISTEMAS ESPECIALISTAS	12
2.1 Sistema Especialista para Diagnóstico de Depressão (Peruzzo et al., 2019)	15
2.2 Ambiental Web: Sistema Especialista para Apoio à Avaliação de Processos de Licenciamento Ambiental (Almeida, Silveira & Cunha, 2015)	21
2.3 Aplicativo Mobile para Avaliar a Acessibilidade de Objetos de Aprendizagem utilizando um Sistema Especialista (Guerra et al., 2020)	26
3 EXPERIÊNCIAS NO DESENVOLVIMENTO DE SISTEMAS DE RECOMENDAÇÃO	34
3.1 Sistema de Recomendação de Produtos de Pet Shop integrado a uma Loja Virtual (Cheffer & Silveira, 2023)	36
3.2 Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom (Colli & Silveira, 2023)	41
3.3 Sistema de Recomendação Mobile: um estudo de caso para delivery (Garbin & Silveira, 2017)	49
3.4 Framework Genérico de Recomendação para Lojas Virtuais (Prado & Silveira, 2014)	54



SUMÁRIO

4 EXPERIÊNCIAS DESENVOLVIDAS COM SISTEMAS HIPERMÍDIA ADAPTATIVOS	60
4.1 Adaptação de Cursos a Distância empregando Objetos de Aprendizagem (Silveira et al., 2016)	63
5 EXPERIÊNCIAS DESENVOLVIDAS COM RACIOCÍNIO BASEADO EM CASOS	68
5.1 Ferramenta para Apoio ao Suporte Técnico em uma Rede FTTX empregando Raciocínio Baseado em Casos (Ferrari et al., 2019)	72
6 EXPERIÊNCIAS DESENVOLVIDAS COM REDES NEURAIIS ARTIFICIAIS	79
6.1 Sistema de Auxílio ao Diagnóstico de Diabetes (Vieira, Parreira & Silveira, 2016)	82
6.2 Aprendendo com o Zag: Jogo Educacional Digital para Alfabetização utilizando Redes Neurais Artificiais (Basso et al., 2016)	87
CONSIDERAÇÕES FINAIS	93
BIBLIOGRAFIA	94
SOBRE OS AUTORES	103





CAPÍTULO 1

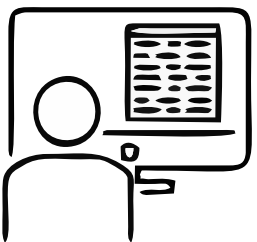
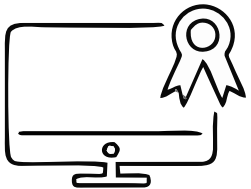
INTELIGÊNCIA ARTIFICIAL E O DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO INTELIGENTES

A

tualmente, a IA (Inteligência Artificial) tem sido tema de muitas discussões, especialmente pelo destaque de ferramentas tais como o *ChatGPT* (OpenAI.com, 2025), uma plataforma de conversação que responde perguntas realizadas em linguagem natural. Vários pesquisadores da área de IA (e também de outras áreas) têm discutido os resultados e os possíveis impactos do uso desta ferramenta que, além de responder perguntas, cria textos e programas de computadores, entre outras possibilidades (Pujol, 2023).

Entretanto, a IA já está presente em nosso cotidiano, sendo presente até mesmo no atendimento eletrônico de várias empresas, seja por meio de contato telefônico, via *Whatsapp* ou por meio de assistentes virtuais. Outras vertentes muito discutidas atualmente compreendem a utilização de ferramentas de IA na área do Direito (para auxiliar no julgamento de processos judiciais, por exemplo) e na área de Educação (tais como ferramentas que possam corrigir atividades desenvolvidas por alunos em um AVA - Ambiente Virtual de Aprendizagem - e o desenvolvimento de Sistemas Tutores Inteligentes) (Fava, 2018; Junquillo, 2022).

A Inteligência Artificial fundamenta-se na ideia de que é possível modelar o funcionamento da mente humana por meio do computador. Segundo Damásio (2012), o corpo proporciona uma referência fundamental para a mente. Segundo este autor, os aspectos neurais e químicos da resposta do cérebro provocam uma alteração profunda no funcionamento dos tecidos e sistemas, tais como: disponibilidade de energia, taxa metabólica, resposta do sistema imunizador e contração da musculatura. Os sinais de todas essas alterações são retransmitidos ao cérebro (cérebro e corpo atualizam constantemente as informações) (Silveira & Vit, 2023).



Neste sentido, não se pode pensar em uma atividade mental plena fora do corpo, visto que o meio ambiente influencia o cérebro por meio das percepções humanas. Quando um ser humano vê, ouve, toca, saboreia, cheira: o corpo e o cérebro interagem juntos com o ambiente. Segundo Damásio (2012), perceber é atuar sobre o meio como também dele receber sinais. A mente surge da atividade nos circuitos neurais, que representam o organismo continuamente, à medida que é perturbado pelos estímulos do meio ambiente físico e sociocultural, e à medida que atua sobre esse meio. Este é um dos pontos críticos da IA, a questão filosófica que envolve corpo, alma e mente (Silveira & Vit, 2023).

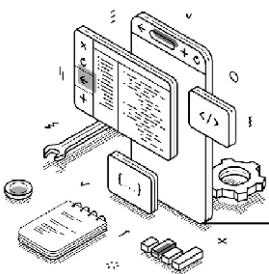
Segundo Lévy (1998), os fundadores da Inteligência Artificial (Herbert Simon, John McCarthy e Marvin Minsky) acreditavam firmemente que a inteligência é um mecanismo. O cérebro, nesta visão, é uma máquina e os neurônios são processadores de informação. Segundo esta visão, de que o cérebro é uma máquina, seria perfeitamente possível modelá-lo dentro do computador (Russell & Norvig, 2021). Mas o mundo em que vivemos, o mundo real, é muito maior e mais complexo do que o micromundo digital do computador, construído por nós mesmos (Silveira & Vit, 2023).

A Inteligência Artificial levanta questões do tipo: Como ocorre o pensar? Como o homem extrai conhecimentos do mundo? Como a memória, os sentidos e a linguagem ajudam no desenvolvimento da inteligência? Como surgem as ideias? Como a mente processa informações e tira conclusões decidindo por uma coisa ao invés de outra? (Lima; Labidi, 1999 *apud* Lorenzi; Silveira, 2011). Essas questões são fundamentais para que se possa simular o raciocínio humano e para que a simulação do funcionamento da mente possa ser implementada por meio do computador (Silveira & Vit, 2023).

Marvin Minsky e Seymour Papert lançaram as bases para o surgimento do paradigma simbólico na IA. Este paradigma aborda a simulação da inteligência não por meio da construção de hardware específico, mas no desenvolvimento de programas computacionais que operam sobre dados ou representações (Teixeira, 1998). Este paradigma envolve a mudança de definição do que é inteligência. Ao invés de definir a inteligência como a capacidade para solucionar problemas, a inteligência é resultante da representação mental, que não é nada mais do que uma atividade simbólica. Segundo Teixeira (1998, p. 44), “O que nos distingue dos outros animais menos inteligentes é nossa capacidade de produzir e manipular símbolos”. Ainda assim, a noção de que a inteligência é a capacidade para resolver problemas continua existindo. Para resolver um problema precisa-se um caminho (um algoritmo) que permita a manipulação adequada de símbolos (atividade simbólica).

Dentro deste paradigma a mente é vista como um processador de informação; esta informação pode ser representada na forma de símbolos; estes símbolos combinam-se entre si por meio de regras. O processamento simbólico envolve a simulação da inteligência por meio da criação de programas computacionais que operam sobre dados ou representações, baseando-se em uma abordagem cognitiva, imitando a forma como o ser humano raciocina (descendente) (Silveira & Vit, 2023).

Além do paradigma simbólico que enfatiza os processos cognitivos, a IA também pode ser abordada do ponto de vista conexionista. Na abordagem conexionista a ênfase é o modelo de funcionamento do cérebro, dos neurônios e das conexões neurais (Lorenzi; Silveira, 2011). Esta abordagem baseia-se nos aspectos biológicos (ascendente). Uma das técnicas de IA que utiliza a abordagem conexionista são as Redes Neurais Artificiais (Haykin, 2007; Silveira & Vit, 2023).



Informação Inteligentes vamos estudar algumas definições para Inteligência Artificial (*Artificial Intelligence*). Existem inúmeras definições. Destacam-se, entre elas:

“...inteligência artificial é um tipo de inteligência produzida pelo homem para dotar as máquinas de algum tipo de habilidade que simula a inteligência do homem” (Lima & Labidi, 1999 apud Lorenzi & Silveira, 2011).

“Uma máquina é inteligente se ela é capaz de solucionar uma classe de problemas que requerem inteligência para serem solucionados por seres humanos” (McCarthy & Hayes, 1969).

“Inteligência artificial é o estudo das faculdades mentais através do uso de modelos computacionais” (Charniak & McDermott, 1985).

“Inteligência artificial é o estudo de como fazer os computadores realizarem coisas que, no momento, as pessoas fazem melhor” (Rich & Knight, 1994).

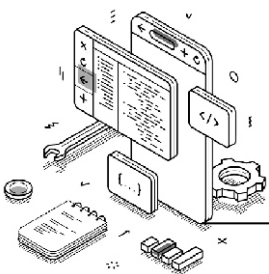
“A Inteligência Artificial pode ser definida como um conjunto de técnicas e metodologias de programação usadas para tentar resolver os problemas de forma mais eficiente que soluções algorítmicas, e fazendo isso o mais próximo possível de um ser humano” (Alves & Pasquareli, 1997 apud Lorenzi & Silveira, 2011).

1.1 Aplicação de Técnicas de IA no Desenvolvimento de Sistemas de Informação Inteligentes

Sistemas de Informação Inteligentes, ao contrário de Sistemas de Informação tradicionais, processam conhecimento, não dados. O conhecimento é volumoso, de difícil caracterização, organizado e é individual. Na área de IA costumamos utilizar conhecimento e não simplesmente dados. Dados são fatos brutos, não processados. O processamento dos dados gera informações. O conhecimento é a interpretação e aplicação dessas informações.

Dados são elementos puros e quantificáveis sobre um determinado evento. Por exemplo, o número de alunos matriculados em um curso de graduação representa um dado. Uma informação envolve a interpretação de um conjunto de dados, por exemplo, comparar as matrículas de acordo com o semestre e verificar se houve ou não aumento no número de alunos. O conhecimento envolve a habilidade de criar um modelo mental que descreva o objeto e indique as decisões que devem ser tomadas. Por exemplo: decidir, de acordo com o número de alunos matriculados, quais turmas/disciplinas devem ou não ser ofertadas no semestre (Rezende, 2005; Silveira & Vit, 2023).

Para desenvolver Sistemas de Informação Inteligentes pode-se aplicar uma técnica de IA, que pode ser compreendida como um método que explora o conhecimento e deve ser representado de tal forma que (Lorenzi & Silveira, 2011):



- Seja compreendido pelas pessoas que o fornecem;
- Seja facilmente modificado;
- Seja usado em inúmeras situações;
- Permita limitar as possibilidades a serem consideradas.

O conhecimento processado por Sistemas de Informação Inteligentes precisa ser representado de forma computacional. Esta representação pode ser realizada de diferentes formas (Fernandes, 2005; Lorenzi & Silveira, 2011; Rezende, 2005; Silveira & Vit, 2023), tais como conhecimento procedural, redes semânticas; *frames*; lógica de predicados; árvores de decisão; conhecimento estatístico; regras de produção; processamento paralelo distribuído (Redes Neurais); raciocínio baseado em casos; e esquemas híbridos (envolvem a utilização de mais de um método de representação do conhecimento de forma combinada).

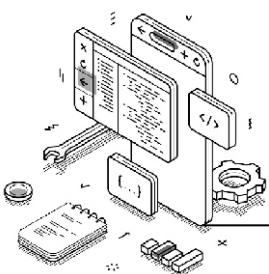
Fávero & Santos (2011 *apud* Lorenzi & Silveira, 2011) apresentam, de forma resumida, as principais diferenças entre o conhecimento humano e o conhecimento artificial (Quadro 1).

Quadro 1: Diferenças entre o conhecimento humano e o conhecimento artificial

Conhecimento Humano	Conhecimento Artificial
Perecível	Permanente
Difícil de transferir	Fácil de transferir
Difícil de documentar	Fácil de documentar
Imprevisível	Consistente
Caro	Razoável
Discriminatório	Imparcial
Social	Individualizado
Criativo	Sem inspiração
Adaptável	Inflexível
Enfoque amplo	Enfoque restrito
Baseado em senso comum	Técnico

Fonte: Fávero & Santos (*apud* Lorenzi & Silveira 2011)

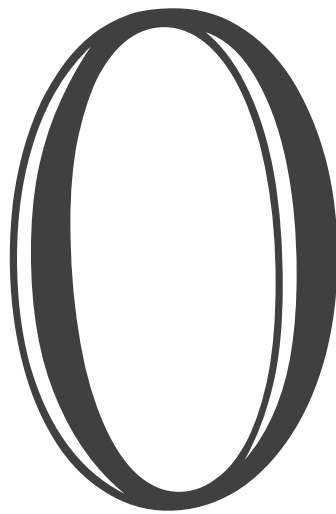
A pesquisa em Inteligência Artificial envolve uma série de subáreas, entre elas os Sistemas Especialistas, Robótica, Redes Neurais Artificiais, Vida Artificial, Sistemas Multiagentes e Algoritmos Genéticos, entre outros (Barone, 2003). Neste livro, serão estudados os Sistemas Especialistas, Sistemas de Recomendação, Mineração de Dados, Sistemas Hipermídia Adaptativos, Algoritmos Genéticos, Raciocínio Baseado em Casos e Redes Neurais. Nos próximos capítulos apresentaremos conceitos de algumas destas áreas, bem como projetos desenvolvidos nas mesmas, pelo grupo de pesquisa IATE-UFSM.





CAPÍTULO 2

EXPERIÊNCIAS NO DESENVOLVIMENTO DE SISTEMAS ESPECIALISTAS

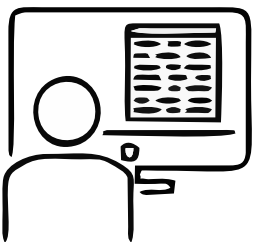
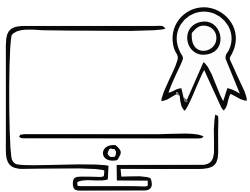


Os SEs (Sistemas Especialistas ou *Expert Systems*) são sistemas desenvolvidos a partir do conhecimento de um ou mais profissionais na área de conhecimento (ou domínio) do sistema proposto. Esses profissionais são os *especialistas do domínio*.

Para que um profissional de Computação desenvolva um SE ele deve ser acompanhado por um especialista do domínio, que o guiará com as informações necessárias conforme sua experiência. Neste contexto, os profissionais de Computação são denominados de Engenheiros do Conhecimento (Lorenzi & Silveira, 2011; Peruzzo *et al.*, 2019).

Os SEs apresentam algumas características que os diferenciam de um Sistema de Informação tradicional, tais como: aumento da eficiência e da flexibilidade, possibilidade para construção de regras e tomada lógica de decisões. Também podem ser destacadas algumas vantagens da utilização de um SE, tais como: velocidade na determinação de problemas, decisão fundamentada em uma base de conhecimento, segurança, pessoas para interagir com o sistema, estabilidade, flexibilidade e integração de ferramentas (Fernandes, 2005; Anselmo & Silveira, 2009). Uma importante característica da programação de SEs é a possibilidade de revisão e mudança, ou seja, a qualquer momento podem ser acrescentadas novas regras e realizados ajustes, mantendo atualizada a base de conhecimento (Luger, 2013). Para isto, o desenvolver do SE (Engenheiro do Conhecimento) deve implementar uma interface que permita a manutenção da base de conhecimento (Peruzzo *et al.*, 2019).

O conhecimento de um SE pode ser representado de diferentes formas, tais como regras de produção e *frames*, por exemplo. Os sistemas baseados em regras “se inspiraram na ideia de que o processo de tomada



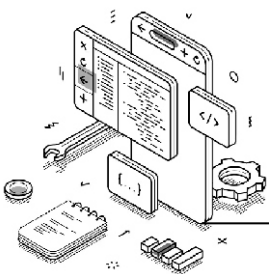
de decisão humana poderia ser modelado por meio de regras do tipo se condições então conclusões e ações” (Rezende, 2005, p. 32).

Já para Zuben (2011), sistemas baseados em regras de produção possuem uma estrutura na forma “SE {antecedente} ENTÃO {consequente}” que relaciona informações ou fatos (no antecedente, também denominado de premissa ou condição) a alguma ação ou resultado (no consequente, também denominado de conclusão). São fáceis de criar e de interpretar. Por meio da base de conhecimento construída com o apoio de um ou mais especialistas, as informações serão verificadas e, por meio dos testes, o SE será capaz de interpretar as respostas (Peruzzo *et al.*, 2019).

Segundo Keller (Keller, 1991 *apud* Lourenço, 2003), a estrutura básica de um SE possui três elementos fundamentais: base de conhecimento, motor (ou máquina) de inferência e interface com o usuário:

- **Base de Conhecimento:** é na base de conhecimento que ficam armazenadas as informações especialistas necessárias para resolver problemas de um domínio específico. Essas informações consistem de fatos e heurísticas, onde os fatos são as informações que estarão sempre disponíveis para o sistema e deverão ser atualizadas pelo especialista humano de acordo com a necessidade e as heurísticas são as regras práticas que caracterizam o nível de tomada de decisão do especialista em um domínio (Chaiben, 2003 *apud* Lourenço, 2003; Peruzzo *et al.*, 2019);
- **Motor de Inferência:** o motor de inferência (ou máquina de inferência), de certo modo, tenta imitar o pensamento que os especialistas humanos empregam quando resolvem um problema, ou seja, ele pode começar com uma conclusão e procurar uma evidência que a comprove, ou pode iniciar com uma evidência para chegar a uma conclusão. Esse tipo de pensamento, nos SEs, está dentro da classe de raciocínio via Regras de Encadeamento e são chamados de *backward chaining* e *forward chaining* respectivamente (Chaiben 2003 *apud* Lourenço, 2003; Peruzzo *et al.*, 2019);
- **Interface com o usuário:** A interface com o usuário deve ser flexível o bastante para que a interação entre o SE e o usuário conduza a uma eficiente navegação na base de conhecimento durante o processamento das heurísticas, permitindo que o usuário descreva o problema ou os objetivos que deseja alcançar e também facilita a recuperação do caminho percorrido pelo sistema para chegar à solução do problema, por meio de um modelo de consulta estruturado. Esse caminho é denominado *trace* e é a base de pesquisa para a explanação, que consiste em explicar “o porquê” e o “como” o sistema chegou a tal conclusão. Esse processo é muito importante, pois oferece ao usuário ajuda para julgar se adota ou não a solução apresentada pelo Sistema Especialista (Mendes 2003 *apud* Lourenço, 2003; Peruzzo *et al.*, 2019).

Como toda e qualquer técnica adotada, existem prós e contras; com os Sistemas Especialistas não é diferente. Desta forma, Zuben (2011) cita vantagens e desvantagens sobre os SEs. Entre as vantagens dos SEs destacam-se: são sistemas concebidos para emular a estratégia de tomada de decisão de



especialistas humanos; tendem a ser muito precisos em suas conclusões; tendem a produzir uma conclusão em um tempo curto, embora testes exaustivos no caso de bases de conhecimento compostas por muitas regras tendem a ser custosos; apresentam uma alta capacidade de explanação do processo de inferência; apresentam uma estrutura uniforme, em que cada regra é um pedaço independente do conhecimento disponível; ao separar o conhecimento do seu processamento, permitem que um mesmo ambiente de projeto de sistema especialista possa ser empregado em diferentes aplicações (Peruzzo *et al.*, 2019).

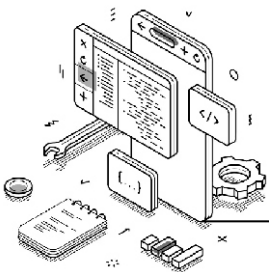
Como desvantagens da aplicação de SEs, Zuben (2011) coloca que: atuam em um domínio restrito; recorrem predominantemente a processos de inferência dedutiva, em detrimento das inferências indutivas e abduativas (forma de inferência que busca a explicação mais provável para um conjunto de observações ou fatos); promovem pouco relacionamento entre as regras, de modo que há poucos subsídios para se definir o papel de regras individuais num processo de inferência; podem cometer erros durante a inferência, possivelmente motivados pela dificuldade de lidar com ambiguidades, pela presença de regras conflitantes na base de conhecimento e pela dificuldade de sofrer atualizações ou aprendizado em sua base de conhecimento (Peruzzo *et al.*, 2019).

Em SEs, a primeira etapa de desenvolvimento compreende a fase de aquisição do conhecimento, que visa a obter o raciocínio do especialista no que se refere a como ele resolve o problema para depois ser inserido na base de conhecimento do sistema. A aquisição do conhecimento não acontece de uma só vez, ela é um processo cíclico e possui quatro etapas: coleta, interpretação, análise e projeto (Lourengo, 2003):

- Coleta: é nesta etapa que o conhecimento do especialista é adquirido. Exige que o profissional da área de TI (denominado Engenheiro do Conhecimento) possua boas ferramentas de intercomunicação pessoal e habilidade para obter a cooperação do especialista;
- Interpretação: as informações coletadas são interpretadas e as peças chave do conhecimento são identificadas;
- Análise: as peças chave do conhecimento são estudadas. Esta etapa relaciona a formação de teorias na organização sobre o conhecimento e estratégias de resolução de problema;
- Projeto: nesta última etapa do ciclo o Engenheiro do Conhecimento já possui um novo conhecimento do problema, mas desse novo conhecimento ele precisa identificar novos conceitos e estratégias que necessitam de maior investigação, para começar um novo ciclo de aquisição do conhecimento.

As principais dificuldades na aquisição do conhecimento, segundo Lourenço (2003) são:

- Inconsciência do conhecimento: o especialista pode se esquecer de informar fatos importantes na descrição dos métodos de resolução de problemas ou fornecer informações inconscientes;



- Dificuldade de verbalizar o conhecimento;
- Aquisição de informações irrelevantes para a resolução dos problemas;
- Conhecimento incompleto: o próprio especialista pode não saber a resolução completa do problema;
- Conhecimento incorreto: as informações fornecidas pelo especialista podem estar incorretas.

Existem várias técnicas para aquisição do conhecimento. O Engenheiro do conhecimento deve especificar para que a mesma seja simples, de rápida aplicação e adaptável a vários domínios. Uma das técnicas mais utilizadas são as entrevistas. Esta técnica é marcada por encontros ou discussões entre o Engenheiro do Conhecimento e o especialista do domínio, objetivando a obtenção de uma visão geral do domínio e o modo de solução de problemas do especialista. As informações podem ser coletadas com o auxílio de gravadores, filmadoras, questionários, ou qualquer outro meio possível, essas informações são posteriormente analisadas para se extrair o conhecimento desejado. Há dois tipos de entrevistas: as não estruturadas e as estruturadas (Lorenzi & Silveira, 2011):

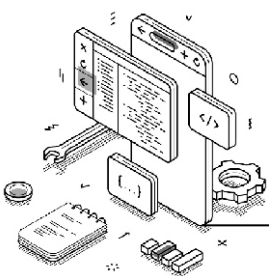
- Não estruturadas: as informações são obtidas por meio de conversa direta com o especialista, na qual ele apresenta o problema e a forma como o mesmo é resolvido. É muito importante interferir o mínimo possível, pois qualquer interferência tende a modificar a linha de raciocínio do especialista;
- Entrevistas Estruturadas: essas entrevistas são mais formais. As perguntas são elaboradas com cuidado e tendem a levar a conversa à aquisição de conhecimentos específicos sobre o domínio. Este tipo de entrevista força o especialista a ser sistemático.

Nas próximas seções serão apresentados alguns Sistemas Especialistas desenvolvidos pelos integrantes do grupo de pesquisa IATE-UFSM.

2.1 Sistema Especialista para Diagnóstico de Depressão (Peruzzo et al., 2019)

Esta seção apresenta o desenvolvimento de um protótipo de SE (Sistema Especialista), construído por Peruzzo *et al.* (2019), possibilitando uma ferramenta que possa auxiliar os profissionais da área de saúde no diagnóstico de depressão, com possíveis sintomas clínicos da doença. A proposta deste trabalho envolveu a implementação de um protótipo de Sistema Especialista (SE) para uso em computadores/notebooks, tablets, smartphones e dispositivos afins, de forma responsiva, via *web*. O protótipo do sistema especialista foi disponibilizado para aplicação dos testes por profissionais da saúde, para validar sua implementação (Peruzzo *et al.*, 2019). O SE foi denominado de SEDD (Sistema Especialista para Diagnóstico de Depressão).

No contexto da Informática na Saúde, esta seção apresenta o desenvolvimento de um sistema que possa apoiar o diagnóstico da depressão. De maneira a



possibilitar uma ferramenta que possa auxiliar os profissionais da área de Psicologia no diagnóstico de depressão nos mais diferentes pacientes com possíveis sintomas clínicos da doença, a proposta deste trabalho envolveu a implementação de um protótipo de SE, disponibilizado apenas aos profissionais da saúde, evitando que qualquer pessoa possa utilizá-lo e receber um diagnóstico errôneo.

A metodologia de pesquisa adotada para o desenvolvimento deste trabalho foi a dissertação-projeto, pois desenvolveu-se um protótipo de SE para auxiliar no diagnóstico de depressão. Segundo Ribeiro & Zabadal (2010), na metodologia de dissertação-projeto, “...o pesquisador caracteriza determinado problema de algum aspecto técnico. Destaca a relevância de resolver esse problema. Desenvolve, então, um programa sistema ou mesmo um protótipo – para apresentar como prova de conceito da solução desse problema” (p. 96).

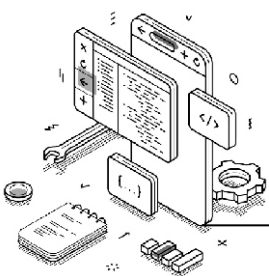
Para que o SE possa auxiliar no diagnóstico de depressão, são aplicados testes, definidos pela especialista do domínio, a Psicóloga Tamara Peruzzo. Após a realização dos testes e a sugestão do diagnóstico pelo SE, os pacientes diagnosticados de forma positiva serão convidados a ter um acompanhamento especializado.

O desenvolvimento do protótipo de SE foi acompanhado, constantemente, pela profissional em Psicologia responsável pelo auxílio no desenvolvimento deste sistema, para que fossem realizadas correções, ajustes de detalhes e, também, para fazer uma validação de como a aplicação pode beneficiar os profissionais desta área.

O SE implementado pode auxiliar os profissionais a descobrir as causas, estimar o tempo em que a doença se agrava, o estágio em que se encontra, e ao final, conforme o diagnóstico dado pelo sistema, orientar a continuidade de acompanhamento ou não do profissional responsável pela realização do teste. O intuito deste sistema é o de facilitar, por meios tecnológicos e práticos, a realização de testes baseados nos sintomas e diagnosticar a presença ou não da doença.

O funcionamento básico do SE se dá por meio de um questionário (anamnese), baseado nos sintomas, sentimentos e emoções, independente do tempo em que se apresenta ou se tenha percebido os sentimentos que se assemelham à depressão. As respostas, bem como o nome do responsável pelo teste, são armazenadas em um banco de dados ou meio disponível e acessível dependendo do local a ser aplicado, para possíveis consultas e acompanhamentos de evolução ou regressão do paciente, em casos mais aprofundados e estágios críticos.

O SE não pretende substituir os profissionais da área de saúde, tampouco receitar medicamentos. A utilização do SE deve ser feita por especialistas da área de saúde, auxiliando-os a diagnosticar a depressão nos pacientes. Inicialmente são contabilizadas as questões respondidas para cada uma das seguintes categorias: depressão, descontentamento com tudo, suicídio, insônia, agitação, ansiedade, sintomas gastrointestinais, impotência sexual, irritabilidade, raiva, desânimo, fraqueza, falta de apetite ou excesso, isolamento social (APA, 2014).



O SE foi desenvolvido para ser utilizado com pacientes que tenham idade maior ou igual a 16 anos. Este critério foi adotado tendo em vista que a partir dessa faixa etária, os pacientes terão melhor entendimento, compreensão e interpretação para responder ao questionário proposto no protótipo do sistema especialista. Também a partir desta idade os sintomas podem ser relacionados a diversos fatores como perda de um ente querido, demissão de um emprego, dificuldade financeira, dificuldade de relacionamento entre outros aspectos, “Uma forma mais crônica de depressão, o transtorno depressivo persistente (distímia), pode ser diagnosticada quando a perturbação do humor continua por pelo menos dois anos em adultos” (APA, 2014).

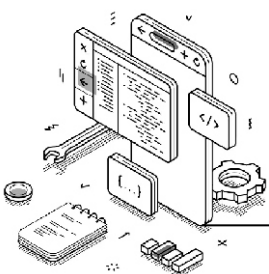
Para a aquisição do conhecimento do SE implementado utilizou-se a técnica de entrevista não estruturada, por meio de vários contatos entre um dos autores deste trabalho (Engenheiro do Conhecimento) e a especialista (Psicóloga Tamara Peruzzo).

Para representar o conhecimento do protótipo do SE implementado, utilizou-se a técnica de *frames*. Segundo Costa e Silva (2005), um *frame* é uma estrutura de dados complexa, capaz de modelar objetos do mundo real. As associações feitas entre as formas determinam suas estruturas hierárquicas. Cada uma dessas associações irá ligar um *frame*-pai ao filho. A hierarquia existente entre os *frames* permite um armazenamento abstrato dos dados, juntando propriedades comuns que são automaticamente herdadas, evitando, assim, a duplicação das informações e simplificando o código, o que proporciona uma fácil manutenção futura do sistema.

Os módulos de aquisição e manutenção do conhecimento possuem o papel de cadastrar todas as informações necessárias para que a máquina de inferência seja capaz de interagir com essas informações, a fim de formar possíveis diagnósticos. No caso do SE implementado, o especialista responsável por construir o conhecimento para o SE deve efetuar o cadastro de todas as características necessárias para auxiliar no diagnóstico da depressão. Após esse cadastro inicial, o especialista deve vincular as informações, atribuindo pesos de importância para cada uma delas. Fazendo isso, esses vínculos criarão possíveis diagnósticos. O núcleo principal desse módulo é a máquina de inferência, responsável por efetuar o cruzamento dessas informações, simulando, assim, a capacidade cognitiva do especialista do domínio. O algoritmo de inferência interage com o conhecimento informado pelo especialista a fim de formar possíveis diagnósticos (Lorenzi & Silveira, 2011).

O processo de inferência do SE implementado é realizado com base em um algoritmo, levando-se em conta as respostas do paciente para o questionário desenvolvido pela especialista. O questionário é composto de uma quantidade de 32 questões, todas com três alternativas cada, excetuando-se a última questão. A última se diferencia das demais com as seguintes alternativas: duas semanas ou mais; durante dois anos ou mais; por mais de duas semanas, porém sinto de quatro a cinco sintomas mais diariamente. Já as demais perguntas do questionário apresentam as seguintes alternativas: sim; não; às vezes.

Os níveis de depressão possíveis de serem diagnosticados são os seguintes: depressão leve, depressão moderada e depressão grave. Para identificar os



níveis de depressão citados, de acordo com as definições da Psicóloga Tamara Peruzzo Stock, o questionário foi desenvolvido e é avaliado da seguinte forma:

- Depressão leve: de 2 a 4 questões respondidas com a alternativa *às vezes*;
- Depressão moderada: pelo menos 3 a 4 sintomas abrangentes e obrigatórios, respondidos com a alternativa *sim* e mais duas outras alternativas dentre 11 possíveis;
- Depressão grave: 5 sintomas obrigatórios mais 4 outros dentre 14 possíveis.

Este questionário foi desenvolvido pela especialista para suprir a necessidade deste protótipo, seguindo como referência o DSM V (Manual Diagnóstico e Estatístico de Transtornos Mentais) (APA, 2014), pois os questionários existentes se limitam apenas ao uso de profissionais da área de Psicologia, passando pela aprovação do órgão regulamentador, não havendo tempo hábil suficiente para a conclusão deste protótipo.

Para o desenvolvimento do protótipo de SE não foram necessários requisitos de hardware específicos. Quanto aos requisitos de software foram utilizados os seguintes componentes: IDE (*Integrated Development Environment*) de desenvolvimento *Netbeans*; Sistema Gerenciador de Bancos de Dados *MySQL*; servidor para hospedagem do protótipo na *web*; servidor *Apache* para executar o SE e navegador *web*. Definiram-se as tecnologias PHP, *MySQL* por questões de domínio da linguagem e por se tratar de melhor adequação das tecnologias empregadas, voltados para a *web*, onde estas possibilitam maiores recursos para estas finalidades.

A Figura 1 apresenta a tela de abertura do SE implementado. Nesta tela é possível visualizar os menus Pacientes, Base de Conhecimento e Anamnese, que permitem o acesso às funcionalidades do SE.

Figura 1: Tela de Abertura do SEDD.



Fonte: Peruzzo *et al.*, 2019

Para que o SE possa auxiliar os profissionais da área de saúde a realizarem os diagnósticos é preciso cadastrar as informações dos pacientes que serão

atendidos. Além disso, a base de conhecimento também precisa ser criada e atualizada. Sendo assim, o SE desenvolvido permite que o especialista do domínio realize o cadastro de novas perguntas e alternativas, assim como pode listar as perguntas cadastradas (para realizar a edição ou exclusão das mesmas). Após a realização dos cadastros iniciais, o especialista pode realizar a anamnese, que permite o acesso ao questionário que será aplicado ao paciente pelo Psicólogo. Ao concluir a anamnese, o Psicólogo deve

clicar em Finalizar Questionário. Neste mesmo menu existe a opção para listar todos os diagnósticos realizados, ordenados pela última data e horário. Ao lado das informações do paciente (Data do diagnóstico, Nome do Paciente e Diagnóstico) existe a opção Detalhes, em que o profissional poderá visualizar as respostas do paciente para cada uma das perguntas (Figura 2).

Figura 2: Tela de Anamnese do SEDD.

Responder Questionário:

Paciente

Selecione o paciente:

1. Sinto-me triste.

Selecione a opção:

2. Sinto-me desesperançoso(a).

Selecione a opção:

3. Pensamentos negativos.

Selecione a opção:

4. Sinto-me vazio.

Selecione a opção:

31. Sensação de peso nas pernas ou nos braços.

Selecione a opção:

32. Baseado nas questões acima, há quanto tempo sente esses sintomas?

Selecione a opção:

FINALIZAR QUESTIONÁRIO

Fonte: Peruzzo *et al.*, 2019

Ao clicar no botão Finalizar Questionário (Figura 2), o SE realiza a inferência, para gerar o diagnóstico. O algoritmo implementa a proposta da especialista do domínio, conforme destacado anteriormente. A Figura 3 apresenta a tela da anamnese com a listagem dos diagnósticos gerados pelo SE para os pacientes.

Figura 3: Tela de Diagnósticos do SEDD.

Diagnósticos Gerados pelo Sistema:

Código	Data do Diagnóstico	Nome do Paciente	Diagnóstico	
1	15/11/2017 21:17:02	Diel Mateus Peruzzo	Leve	DETALHES

© Diel Mateus Peruzzo - Todos os Direitos Reservados

Fonte: Peruzzo *et al.*, 2019

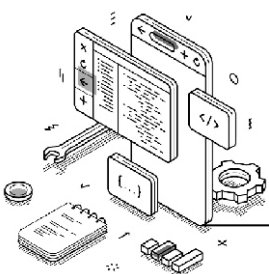
O protótipo do SEDD foi testado e validado. Primeiramente foram testadas as funcionalidades do SE, cadastrando, editando e excluindo dados fictícios de pacientes, perguntas e alternativas, verificando se as mesmas estavam sendo cadastradas corretamente no banco de dados e nas listagens verificando se as mesmas não apresentavam erros de incompatibilidade com o padrão do banco de dados utilizado, já que é bastante comum ocorrerem erros de acentuação se o banco de dados não for configurado de forma adequada (no padrão UTF-8, padrão de reconhecimento de acentos e caracteres especiais em bancos de dados).

Em um segundo momento foi implementado o algoritmo de inferência do SEDD. Após sua conclusão, foram testadas todas as possibilidades, para identificar se o resultado gerado ficou dentro dos quatro possíveis diagnósticos sendo eles: Leve, Moderado e Grave e Indefinido. Caso o mesmo não se encaixe nas regras do algoritmo de inferência, ou seja, casos mais específicos, a decisão final do resultado fica a cargo do Psicólogo que estiver aplicando o questionário ao paciente. Ao finalizar o questionário verificou-se se o diagnóstico era armazenado de forma correta no banco de dados. Foram elaboradas e testadas, juntamente com a especialista que acompanhou o desenvolvimento do SE, diversas combinações de respostas para os diagnósticos possíveis, até que se chegou a uma solução genérica e que possa abordar um conjunto de respostas para gerar os diagnósticos.

Na etapa de validação foram realizados os testes pelas profissionais convidadas para a validação do protótipo de SE desenvolvido (contando com a participação de quatro profissionais da área de Psicologia), a fim de verificar se todas as suas funcionalidades estavam funcionando corretamente, bem como responder ao questionário e posteriormente obter os resultados dos diagnósticos. Os questionários foram realizados baseados em casos reais e alguns aplicados a pacientes, a fim de aferir resultados dentre os diagnósticos gerados pelo SE. Nos testes realizados pelas especialistas obtivemos com sucesso os resultados dos diagnósticos. Os autores buscaram, também, nesta etapa, verificar se o sistema atende e auxilia de forma adequada o profissional que está aplicando o teste aos pacientes, além de buscar a melhoria em seu desenvolvimento por meio da validação do profissional que nos auxiliou na construção do mesmo.

Com relação à implementação do protótipo do SE, o mesmo foi desenvolvido, testado e validado, obtendo as respostas esperadas tanto nas funcionalidades do sistema, assim como na implementação do algoritmo responsável pela verificação das respostas obtidas na anamnese, gerando desta forma os diagnósticos esperados. Existem melhorias a serem feitas no algoritmo para casos específicos. No caso de respostas como pensamentos suicidas ou em casos extremos, podem ser destacadas as respostas, para que o profissional que está realizando a anamnese tomar outras medidas de prevenção, zelando pela vida do paciente.

Entre as dificuldades, destacam-se os poucos SEs existentes na área da depressão, dificultando as comparações de sistemas com tecnologias atuais. Além disso, o acesso aos questionários existentes e sua liberação pelos órgãos responsáveis e que detém os direitos autorais também foi uma dificuldade



encontrada. Desta forma, foi preciso solicitar que a especialista desenvolvesse um questionário próprio, para ser utilizado no desenvolvimento do protótipo.

Como trabalhos futuros, pode ser implementado um algoritmo de inferência mais específico para casos de pacientes com respostas extremas, com um número maior de possibilidades ou desabilitar/preencher outras questões a serem definidas pelos especialistas. Na validação também foi sugerida a inclusão de *links* ou botões no sistema para sugestões de entidades, órgãos ou profissionais autônomos (Psicólogos) que estejam usando o sistema, bem como a localização em que os mesmos se encontram.

2.2 **Ambiental Web: Sistema Especialista para Apoio à Avaliação de Processos de Licenciamento Ambiental (Almeida, Silveira & Cunha, 2015)**

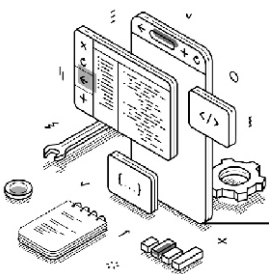
Esta seção apresenta o desenvolvimento de um protótipo de SE para apoiar a avaliação de processos de licenciamento ambiental, denominado de *Ambiental Web* (Almeida, Silveira & Cunha, 2015).

O processo de licenciamento ambiental é um procedimento administrativo que mede o impacto que um empreendimento poderá trazer ao meio ambiente, buscando ao máximo a redução dos impactos negativos que as atividades utilizadoras de recursos naturais e potencialmente poluidoras ou aquelas que, sob qualquer forma, possam causar degradação ambiental (Ministério do Meio Ambiente, 2009).

Neste contexto foi desenvolvido o *Ambiental Web*, uma ferramenta que auxilia na análise prévia e automatizada do processo de corte eventual de árvores, instituindo na sua base de conhecimento princípios, fixando objetivos e normas básicas para a proteção do meio ambiente, estabelecidos na lei, de cada localidade, observando as demais normas estaduais e federais.

O *Ambiental Web* é um protótipo de SE, visando auxiliar na análise de informações, proporcionando mais agilidade aos profissionais da área ambiental com especial interesse na obtenção de licenças ambientais florestais, seja em propriedades particulares ou em áreas de passeio público. As informações que compõem a base de conhecimento do SE compreendem as legislações ambientais nas esferas Municipal, Estadual e Federal, sob o ponto de vista e conhecimento do analista técnico ambiental.

O SE implementado tem, como dados de entrada, informações pessoais do solicitante, localização da propriedade, licenciamento ambiental e espécies que serão cortadas. O SE, a partir da inferência em sua base de conhecimento, informa em quais situações são permitidas o corte de determinada árvore, quais espécies e quantidade de mudas a serem repostas, apoiando o especialista na tomada decisão de aprovar ou não o pedido. Na implementação do protótipo os autores trataram apenas de um tipo de requerimento, que é a emissão de Alvará de Licenciamento Florestal para corte eventual de árvores.



Segundo o Departamento de Meio Ambiente da Prefeitura de Frederico Westphalen-RS, este tipo de requerimento corresponde a mais de 50% da demanda de pedidos de licença.

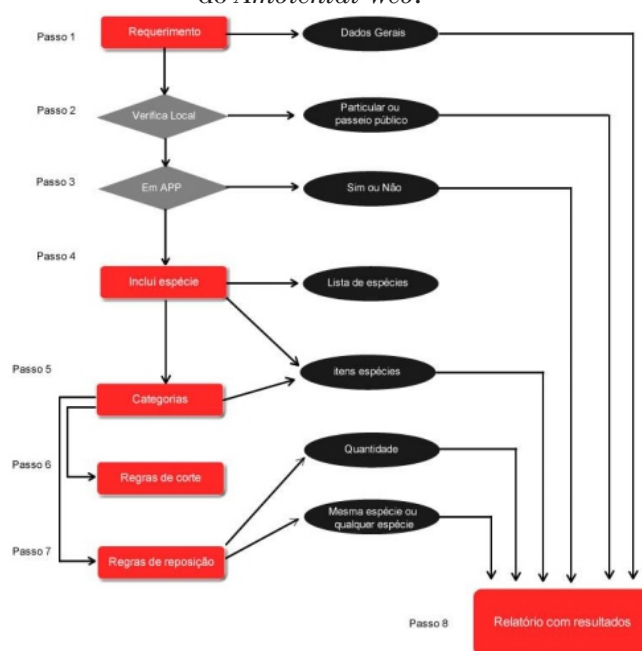
O *Ambiental Web* apresenta uma interface de acesso por meio de um navegador *web*, onde o administrador (ou os técnicos ambientais), realiza a manutenção da base de conhecimento e definição das regras do SE. Também é possível acessar todos os requerimentos e resultados da inferência.

Para facilitar o entendimento da área do conhecimento envolvida, os autores contaram com um especialista, Técnico do Departamento de Meio Ambiente, responsável pela fiscalização, laudos e licenças ambientais na Secretaria de Meio Ambiente de Frederico Westphalen/RS, que acompanhou o desenvolvimento do *Ambiental Web*. Com o apoio deste especialista, foram criados fluxogramas que delimitaram o domínio do sistema, permitindo a construção da base de conhecimento e definição do mecanismo de inferência do SE.

A aquisição do conhecimento foi realizada por meio de reuniões e entrevistas informais com o técnico especialista, além de consultas às leis que compõem a legislação ambiental nos níveis Federal, Estadual e Municipal. Por meio desses encontros foi possível entender a dinâmica dos processos de licenciamentos ambientais e a importância de tais mecanismos de proteção do meio ambiente.

O processo de inferência foi implementado no código-fonte do SE, por meio da consulta às regras definidas na base de conhecimento. O especialista ou o usuário final, a partir do cadastro dos requerimentos, dispara o processo de inferência. Esse processo funciona por meio da comparação de todos os dados recebidos no requerimento, com as regras cadastradas. A Figura 4 apresenta o fluxograma genérico com os passos do processo de inferência.

Figura 4: Fluxograma para Construção do Processo de Inferência do *Ambiental Web*.

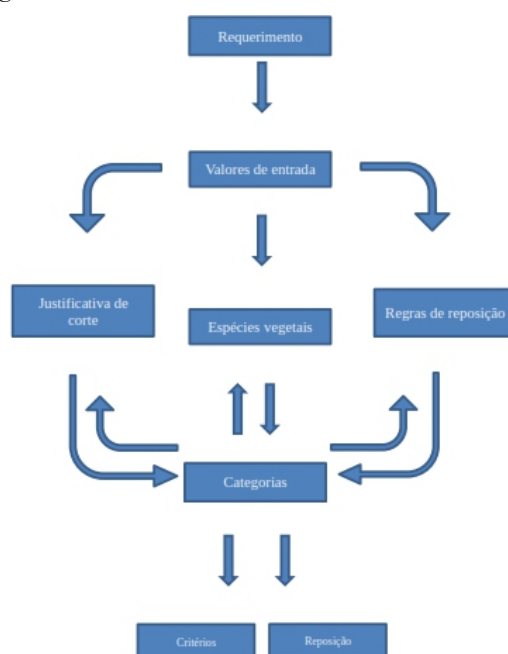


Fonte: Almeida, Silveira & Cunha, 2015

As regras são compostas de itens pré-definidos pelo especialista que, a partir do seu conhecimento técnico, informa ao sistema quais espécies de árvores são vulneráveis, quais requisitos são necessários, a lei e as justificativas que dão suporte ao corte, a quantidade e tipos de espécies de mudas que devem ser repostas. Como ponto de partida para implementar o processo de inferência, os autores utilizaram os fluxogramas (roteiros) elaborados em conjunto com o especialista.

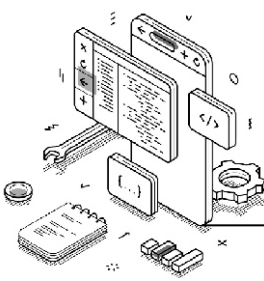
Os fluxogramas/roteiros criados junto com o especialista permitem a identificação dos componentes das regras e como as mesmas se relacionam, para que sejam cadastradas na base de conhecimento. Após a inserção das regras na base de conhecimento, a partir da entrada de pedidos de licenciamento ambiental e verificação dos resultados fornecidos pelo SE, podemos verificar se os resultados estão de acordo com o esperado. A Figura 5 apresenta, de forma gráfica, como funciona o processo de inferência no *Ambiental Web*.

Figura 5: Processo de Inferência do *Ambiental Web*.



Fonte: Almeida, Silveira & Cunha, 2015

Para disparar o processo de inferência do SE, é preciso o preenchimento dos dados pessoais do solicitante. Caso exista licença anterior, deve ser informada a data da última solicitação e o volume de madeira extraído, informar se a propriedade é particular ou se é passeio público e se essa propriedade encontra-se em APP (Área de Preservação Permanente) - Reserva Legal ou Área de Preservação Permanente. Reserva legal é a área localizada no interior de uma propriedade ou posse rural, necessária ao uso sustentável dos recursos naturais e, as APPs constituem áreas protegidas, cobertas ou não por vegetação nativa, com a função ambiental de preservar os recursos hídricos, a paisagem, a estabilidade geológica, a biodiversidade, o fluxo gênico de fauna e flora, além de proteger o solo e assegurar o bem-estar das populações humanas (BRASIL, 2012). Após, o solicitante deve informar, individualmente, quais as espécies de árvores serão solicitadas para corte. Essas espécies serão selecionadas em uma base pré-cadastrada pelo especialista.

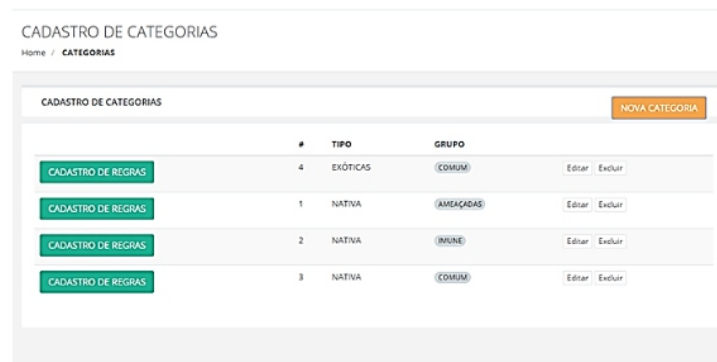


Com base nessas informações, o SE verifica uma a uma as espécies solicitadas com as suas respectivas regras legais ou técnicas que justificam o corte e com as regras de reposição de mudas, levando-se em conta se necessitam ser da mesma espécie ou se podem ser de espécies variadas para, ao final do processo de inferência, apresentar o resultado referente ao requerimento.

Para desenvolver o *Ambiental Web* foi utilizada a linguagem de programação PHP, o gerenciador de Banco de dados *Mysql* para armazenar a base de conhecimento e o *framework Bootstrap*, que contém diversos componentes visuais desenvolvidos com CSS (*Cascade Style Sheet*) e bibliotecas implementadas em *JavaScript*. Para instalar o SE, faz-se necessário um servidor *web* que suporte as tecnologias citadas.

Para demonstrar as funcionalidades implementadas no *Ambiental Web*, foram escolhidas algumas das telas que apresentam as principais tarefas que podem ser desenvolvidas com apoio do SE implementado. A Figura 6 apresenta a tela de *Cadastro de Categorias*.

Figura 6: Tela de Cadastro de Categorias no *Ambiental Web*.

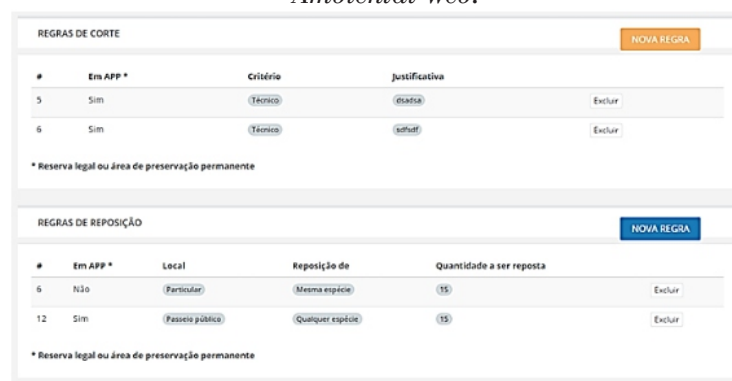


#	TIPO	GRUPO
4	EXÓTICAS	COMUM
1	NATIVA	AMEAÇADAS
2	NATIVA	INUNE
3	NATIVA	COMUM

Fonte: Almeida, Silveira & Cunha, 2015

Como mostra a Figura 6, o especialista cria as categorias que serão usadas para agrupar e ligar as regras que vão definir as justificativas de corte e os critérios de reposição de espécies. A Figura 7 apresenta a interface para que o especialista faça o cadastro de regras de corte e de reposição de árvores.

Figura 7: Cadastros das Regras de Corte e Reposição de Árvores no *Ambiental Web*.



#	Em APP *	Critério	Justificativa
5	Sim	Técnico	clada
6	Sim	Técnica	cultrif

* Reserva legal ou área de preservação permanente

#	Em APP *	Local	Reposição de	Quantidade a ser reposta
6	Não	Particular	Mesma espécie	15
12	Sim	Passivo pública	Qualquer espécie	15

* Reserva legal ou área de preservação permanente

Fonte: Almeida, Silveira & Cunha, 2015

Por meio das funcionalidades apresentadas na Figura 7, o especialista cria as regras de justificativas de corte e os critérios de reposição de espécies conforme a legislação e seu conhecimento técnico. As regras de corte são justificadas com base legal, destacando em que hipóteses é admitido o corte de determinada categoria de árvore. Com relação às regras de reposição, são definidas quantas mudas deverão ser repostas e se deverão ser da mesma espécie ou se podem ser de espécies variadas. Além desses cadastros, a base de conhecimento também precisa ser alimentada com as espécies de árvores, devendo ser informados os nomes popular e científico da espécie, o fator de forma de cada espécie, valor que é usado para o cálculo de volume e a qual categoria pertence a espécie.

Após a realização dos cadastros relacionados à manutenção da base de conhecimento (categorias, regras de reposição, regras de corte e espécies), o solicitante do licenciamento ambiental pode inserir um novo requerimento. A partir dos requerimentos é disparado o processo de inferência do SE.

Por meio de um formulário, o requerente informa os dados pessoais e de localização da propriedade onde estão as espécies que pretende cortar. Após, é preciso adicionar as espécies de árvores que serão cortadas, como mostra a Figura 8.

Figura 8: Requerimento: espécies de árvores.

ADICIONAR ESPÉCIE

Selecione a espécie: DAP: Altura: ADICIONAR ESPÉCIE

ESPÉCIES DE ÁRVORES

Nome popular / científico	DAP	Altura	volume	Regra	fator	
Araucária / Araucária angustifolia	0.85	20	9.92534375	NATIVA - AMEAÇADAS	0.875	Excluir
Grápia / Apuleia leiocarpa	0.78	12	5.01473700000001	NATIVA - AMEAÇADAS	0.875	Excluir
Angico vermelho / Parapiptadenia rigida	0.5	12	2.060625	NATIVA - COMUM	0.875	Excluir

* Reserva legal ou área de preservação permanente

Gravar solicitação Finalizar solicitação

Fonte: Almeida, Silveira & Cunha, 2015

Para informar as espécies, o usuário deve adicionar as espécies de árvores para corte, com base no cadastro de espécies que já estão pré-cadastradas na base de conhecimento. É preciso informar a espécie, o DAP (diâmetro na altura do peito) e a altura de cada árvore. Após o preenchimento de todos os dados referentes ao requerimento é disparado o processo de inferência, que irá gerar um resultado, na forma de relatório, apresentado na Figura 9.

Figura 9: Relatório com os Resultados do Processo de Inferência

RELATÓRIO COM RESULTADOS DA CONSULTA

Home / RESULTADO

Dados do Requerente	Data	Requerimento	Em APP	Propriedade
Luciano rosa de almeida CPF: 12.312.312-3-47 Av João Muniz Reis, 1522 - Frederico westphalen	24-06-2015	Nº 2	Sim	Particular

Espécies solicitadas para corte				
Espécie	DAP	Altura	Volume	Tipo / Grupo
Grápia / Apuleia leiocarpa	0.78	12	5.014737	NATIVA - AMEAÇADAS
Angico vermelho / Parapiptadenia rigida	0.5	12	2.060625	NATIVA - COMUM
Araucária / Araucária angustifolia	0.85	20	9.92534375	NATIVA - AMEAÇADAS

Fonte: Almeida, Silveira & Cunha, 2015

O relatório (Figura 9) apresenta as informações de dados pessoais do solicitante, localização da propriedade e a lista de espécies solicitadas para corte. A Figura 10 apresenta a continuação do resultado da inferência do SE, trazendo as justificativas com as regras de corte e as regras de reposição, incluindo a quantidade e quais espécies de mudas devem ser repostas.

Figura 10: Detalhamento dos Resultados do Processo de Inferência.

REGRAS DE CORTE	REGRAS DE REPOSIÇÃO
Para as seguintes espécies 'NATIVA - AMEAÇADAS' é necessário obedecer as seguintes regras para corte: Crêterios Legal: Implantação de obras ou atividades de interesse público ou social Conforme descrito na legislação supra citada existe previsão legal para manejo.	Para as seguintes espécies NATIVA - COMUM é necessário reposição das seguintes espécies: 15 mudas de qualquer espécie
Para as seguintes espécies 'NATIVA - COMUM' é necessário obedecer as seguintes regras para corte: Crêterios Legal: Risco de queda Com base na Lei Federal 12651	Para as seguintes espécies NATIVA - AMEAÇADAS é necessário reposição das seguintes espécies: 15 mudas de Gráphia (Apuleia lelocarpa)
Crêterios Legal: Implantação de obras ou atividades de interesse público ou social Lei Federal 12651, Lei do Estado Rio grande do Sul - 9519	Para as seguintes espécies NATIVA - AMEAÇADAS é necessário reposição das seguintes espécies: 15 mudas de Araucária (Araucária angustifolia)

Fonte: Almeida, Silveira & Cunha, 2015

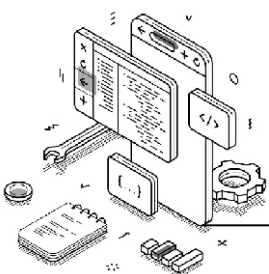
Após a implementação do *Ambiental Web*, foram realizados testes em situações reais do cotidiano. Para tanto, os técnicos ambientais da Prefeitura de Frederico Westphalen/RS buscaram resultados de pareceres redigidos de forma manual e compararam com os resultados do sistema. Para refinar os resultados, os especialistas puderam modificar regras já existentes na base de conhecimento, além de criar novas regras, até conseguirem um resultado satisfatório.

Entre as dificuldades encontradas, destaca-se o pequeno número de SEs existentes na área ambiental. Como existem poucos exemplos de SEs implementados no domínio proposto, os autores precisaram se basear em outros domínios que utilizam as mesmas técnicas para realizar um estudo comparativo.

O *Ambiental Web* traz, como principal benefício, o acesso a informações de forma rápida e precisa para que, mesmo que com pouca experiência, técnicos e profissionais com interesse e conhecimento do domínio, consigam avaliar sem dificuldades a maior parte das questões com relevância sobre os processos de licenciamentos florestais.

2.3 Aplicativo Mobile para Avaliar a Acessibilidade de Objetos de Aprendizagem utilizando um Sistema Especialista (Guerra et al., 2020)

Esta seção apresenta o desenvolvimento de um aplicativo *mobile* para avaliar a acessibilidade de OAs (Objetos de Aprendizagem), aplicando técnicas de Inteligência Artificial, mais especificamente um Sistema Especialista (SE). O



trabalho foi desenvolvido por Guerra *et al.* (2020). O objetivo principal foi o desenvolvimento de um protótipo que avalia a acessibilidade de OAs no formato de texto, inicialmente nos formatos DOCX (*Office Open XML Format*) e *ePub* (*Electronic Publication*), com base em diretrizes e recomendações que são armazenadas na base de conhecimento do SE. Como resultados, o aplicativo permite o cadastro de diferentes OAs e gera um relatório sobre os mesmos, indicando se as recomendações de acessibilidade estão ou não sendo cumpridas.

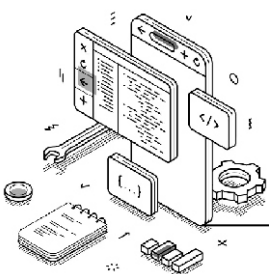
Atualmente, as TDICs (Tecnologias Digitais da Informação e da Comunicação) estão auxiliando professores e alunos nos processos de ensino e de aprendizagem, principalmente no que diz respeito à utilização de materiais didáticos-digitais (também conhecidos como Objetos de Aprendizagem - OAs) e na EaD (Educação a Distância). Diferentes OAs podem ser empregados para apoio às atividades de aula, em diferentes níveis de ensino e em diferentes modalidades (incluindo a modalidade presencial) (Pereira, Bertagnolli & Silveira, 2017; Parreira, Falkembach & Silveira, 2018). Outra tendência no mundo digital é a de ampliar a acessibilidade, permitindo que pessoas que possuem diferentes deficiências possam ter acesso à informação.

A motivação para o desenvolvimento deste trabalho encontra-se nesse contexto, aliando a utilização das TDICs como apoio aos processos de ensino e de aprendizagem (por meio da aplicação de OAs) e a acessibilidade, pois foi desenvolvido um protótipo de aplicativo *mobile* que permite verificar se os OAs existentes e disponíveis na *web* são acessíveis, para que possam ser utilizados como materiais didáticos-digitais para as pessoas com deficiência.

A verificação dos OAs é realizada por meio da análise do cabeçalho dos arquivos nos formatos DOCX e *ePub*. A análise é realizada por meio de um SE. Este SE possui uma base de conhecimento, contendo diretrizes e recomendações que devem ser seguidas para que um OA seja acessível. A base de conhecimento do SE foi organizada em *frames* (ou quadros), que são uma estrutura de informação que consiste em um conjunto de atributos que, por meio de seus valores, descrevem as características de um OA acessível (Lorenzi & Silveira, 2011)

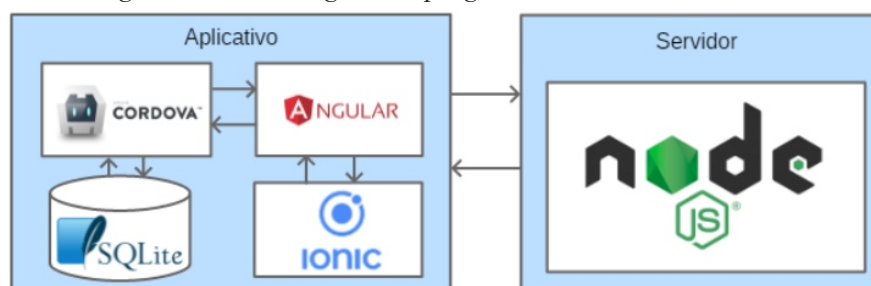
O trabalho faz parte de um esforço de pesquisa conjunto, unindo o grupo IATE/UFSC e o NADTA/UFSC (Núcleo de Acessibilidade Digital e Tecnologias Assistivas/Universidade Federal de Santa Catarina) (NADTA, 2019).

O desenvolvimento do aplicativo ocorreu por meio da utilização do *framework Ionic*, que possibilita compilar e/ou empacotar o aplicativo para utilização em diferentes sistemas operacionais, tais como *Android* e *iOS*. Vilete e Lopes (2018) destacam que o *Ionic* é um *framework* para desenvolvimento de aplicações móveis híbridas, com código aberto e amplamente utilizado. Permite a utilização de HTML (*Hypertext Markup Language*), CSS (*Cascading Style Sheets*) e a linguagem de programação *Javascript* nos projetos, além de ferramentas tais como *Apache Cordova* (realiza a conversão/compilação do projeto para os Sistemas Operacionais *Mobile*) e *Angular*. A metodologia de pesquisa utilizada para o desenvolvimento deste trabalho foi a Dissertação-Projeto, pois se desenvolveu um protótipo do aplicativo (Ribeiro & Zabadal, 2010).



A avaliação da acessibilidade dos OAs é realizada por meio de um SE desenvolvido nas linguagens de programação *Javascript* e *TypeScript*, em conjunto com o *framework Angular*. O SE analisa os parâmetros extraídos dos OAs e compara o conteúdo com a base de conhecimento do SE, possibilitando, assim, mensurar a acessibilidade do OA. A base de conhecimento do SE foi organizada em *frames* (ou quadros), que são uma estrutura de informação que consiste em um conjunto de atributos que, por meio de seus valores, descrevem as características de um OA acessível (Lorenzi & Silveira, 2011). Cada *frame* armazena as características dos OAs, tais como: tipo de mídia utilizado, deficiência atendida, diretrizes e recomendações que foram seguidas no seu desenvolvimento, etc. As tecnologias utilizadas no desenvolvimento do aplicativo compreendem: linguagem de programação *JavaScript*, *framework Angular*; linguagem de programação *TypeScript*; *Apache Cordova*; *SQLite*; e *Node.js*. A Figura 11 apresenta a interação que ocorre entre as tecnologias empregadas no aplicativo e no servidor. O servidor hospeda a API (*Application Program Interface*) que realiza a extração dos parâmetros dos Oas.

Figura 11: Tecnologias Empregadas no Desenvolvimento.

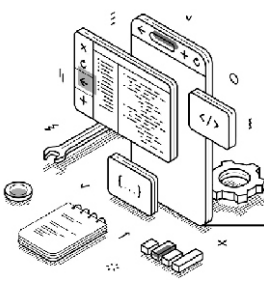


Fonte: Guerra *et al.*, 2020

Inicialmente, o protótipo do aplicativo analisa OAs nos formatos *DOCX* e *ePUB*, por meio da análise do cabeçalho destes arquivos. Conforme Araújo *et al.* (2013) o formato *ePUB* é o formato padrão de publicações digitais, que possui padrão aberto e que utiliza *HTML5*, *CSS*, *SVG* (*Scalable Vector Graphics*), imagens e outros recursos, para representar o conteúdo. O formato *DOCX* é o formato atual dos documentos elaborados por meio do *Microsoft Office*.

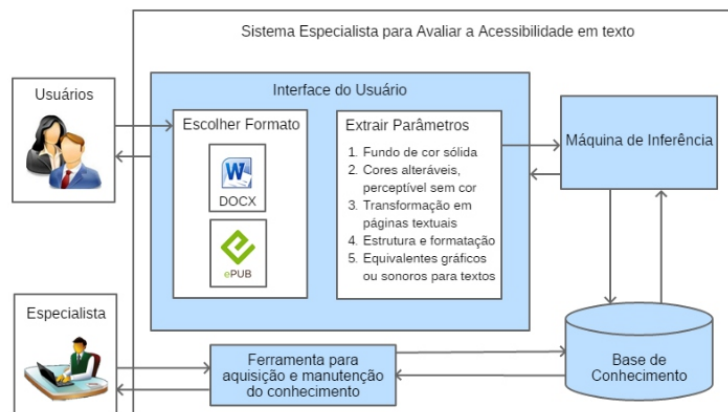
As recomendações de acessibilidade que foram seguidas no desenvolvimento do SE correspondem ao padrão internacional de Diretrizes de Acessibilidade para Conteúdo Web, *WCAG* (*Web Content Accessibility Guidelines*), versão 2.0 (2014), desenvolvido e mantido pela *W3C* (*World Wide Web Consortium*), principal entidade responsável pela padronização na *World Wide Web* (W3C, 2014), além das recomendações e diretrizes propostas por Macedo (2010).

A Figura 12 apresenta uma arquitetura de alto nível do aplicativo desenvolvido. A parte *Extrair Parâmetros* é realizada por meio de uma API desenvolvida, anteriormente, por demais integrantes do Núcleo de Acessibilidade Digital e Tecnologias Assistivas da UFSC (NADTA, 2019). Esta API recebe o arquivo (OA) enviado pelo aplicativo e armazena-o temporariamente no servidor para extrair os parâmetros, retornando-os para o aplicativo. Os parâmetros extraídos correspondem a uma organização



hierárquica do tipo árvore, das *tags* extraídas dos arquivos nos formatos DOCX e *ePub*. Os parâmetros extraídos são salvos em um documento JSON que é lido pelo aplicativo desenvolvido.

Figura 12: Arquitetura de Alto Nível do Aplicativo.



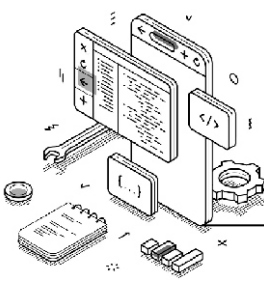
Fonte: Guerra *et al.*, 2020

O SE foi desenvolvido com base nos seguintes requisitos: 1) uma interface que permite a manutenção do cadastro dos OAs e das diretrizes e recomendações de acessibilidade, gerenciada pelo administrador do sistema; 2) uma interface que permite aos usuários submeterem os OAs para verificação da acessibilidade. Nesse sentido, a ideia foi a de criar um SE que possa ser utilizado em diferentes tipos de OAs, que utilizam diferentes mídias (textos, imagens estáticas, animações, tabelas, vídeos, áudio, etc). Apesar de o protótipo, inicialmente, só verificar a acessibilidade de textos, o SE foi modelado para armazenar conhecimento sobre OAs com outras mídias, visando sua flexibilidade de uso.

A máquina de inferência do SE verifica se todas as diretrizes e recomendações, de acordo com o tipo de mídia e deficiências, estão sendo cumpridas, trazendo, como resultado, quais regras foram ou não cumpridas. As diretrizes adotadas no desenvolvimento do aplicativo seguem a proposta de Macedo (2010). A máquina de inferência faz a comparação dos atributos que estão armazenados na base de conhecimento, de acordo com as diretrizes e recomendações, com os parâmetros extraídos dos OAs, por meio da API externa mencionada anteriormente (NADTA, 2019).

O SE desenvolvido possui uma base de conhecimento, na qual são armazenadas as características dos OAs (mídias, deficiências, diretrizes, recomendações): uma máquina de inferência (que verifica se os OAs cumprem as diretrizes propostas, estruturada na forma de *frames*), um subsistema de aquisição de conhecimento (que compreende a manutenção dos cadastros, que só pode ser realizada pelo administrador do SE – no caso, especialistas em acessibilidade) e a interface com o usuário (que submete OAs para serem avaliados pelo aplicativo desenvolvido).

Para o desenvolvimento do aplicativo, que analisa OAs nos formatos DOCX e *ePub*, foram utilizadas, para compor a base de conhecimento, as diretrizes de acessibilidade propostas por Macedo (2010).



A Figura 13 exibe a tela inicial do aplicativo, que apresenta várias informações em relação ao seu propósito e funcionamento, além de informar um e-mail para envio de problemas, sugestões e demais comentários sobre a usabilidade do aplicativo.

Figura 13: Tela Inicial do Aplicativo.



Fonte: Guerra *et al.*, 2020

A Figura 14 apresenta informações e instruções sobre a seleção de OAs para avaliação pelo aplicativo, informando os dois formatos aceitos atualmente, que são: DOCX e *ePub*. O aplicativo realiza uma validação prévia com base no MIME *type* do arquivo. O MIME *type* é um padrão que descreve o tipo de conteúdo de um arquivo, permitindo que tanto clientes como servidor consigam saber qual ação tomar, com base no tipo do arquivo. É utilizado no protocolo HTTP e em protocolos de e-mail, entre outros. Sua estrutura geral compreende duas *Strings* separadas por uma barra '/': a primeira parte informa um tipo discreto, como *text*, *image*, *audio* ou *multipart* que indica que o documento que contém o arquivo possui múltiplas partes; a segunda parte corresponde a um subtipo, como *image/jpeg* para indicar uma imagem com formato JPEG. Os MIME *types* utilizados no aplicativo são os que representam os arquivos com extensão DOCX e *ePub*, respectivamente: *application/epub+zip* e *application/vnd.openxmlformats-officedocument.wordprocessingml.document*.

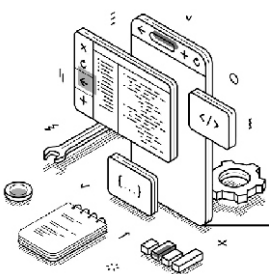


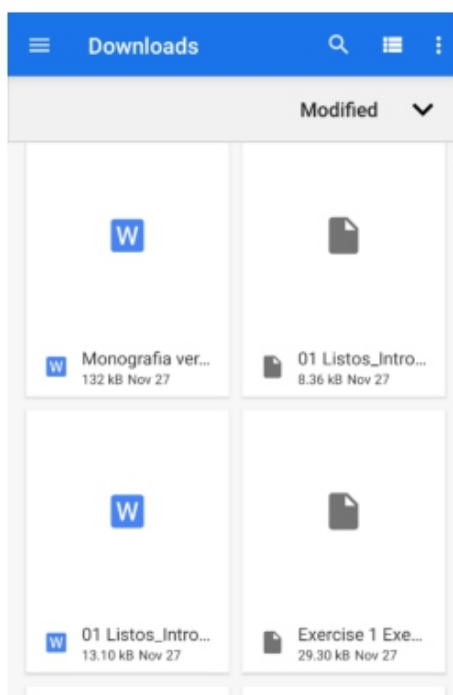
Figura 14: Tela de Seleção do Arquivo.



Fonte: Guerra *et al.*, 2020

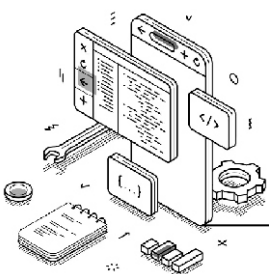
A Figura 15 exibe a tela de seleção nativa do SO *Android*, que permite selecionar apenas arquivos nos formatos válidos.

Figura 15: Tela de Escolha do Arquivo no *Android*.



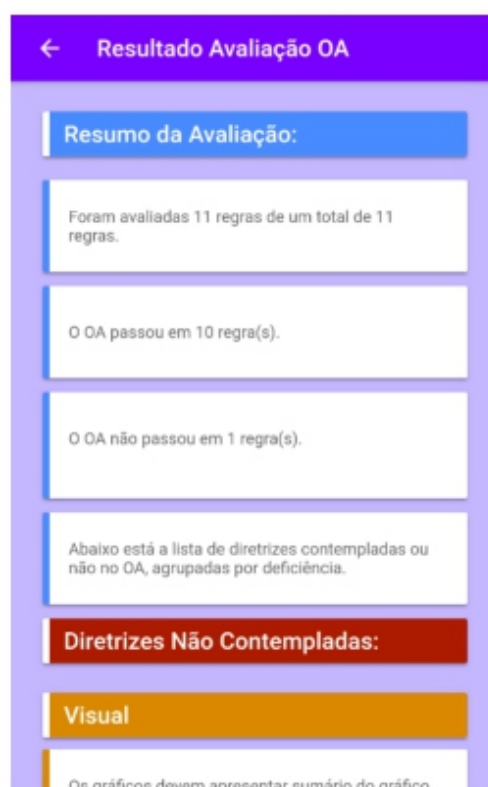
Fonte: Guerra *et al.*, 2020

A Figura 16 representa um exemplo de resultado de avaliação de um OA, apresentando informações tais como: regras/recomendações avaliadas: total de regras que se aplicam ao conteúdo e ao formato do OA; quantidade de regras/recomendações que o OA atende; correspondendo a um contador das regras em que não foram encontrados problemas de acessibilidade na avaliação, conforme as diretrizes; quantidade de regras que o OA não atende:



representando um contador das regras não atendidas, que apresentam problemas de acessibilidade conforme as diretrizes; diretrizes contempladas: compreendem todas as diretrizes em que não foram encontrados problemas, agrupadas por deficiência; diretrizes não contempladas: compreendem todas as diretrizes não atendidas, que apresentam problemas de acessibilidade, também agrupadas por deficiência.

Figura 16: Resultado do Processo de Inferência do SE no aplicativo.

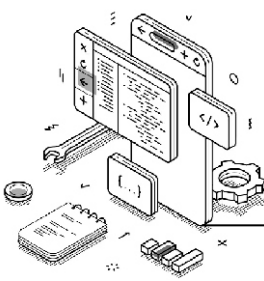


Fonte: Guerra *et al.*, 2020

Após a implementação do protótipo do aplicativo, o mesmo foi testado com base em OAs dos tipos DOCX e *ePub*, localizados nos repositórios MERLOT (MERLOT.ORG, 2019) e Manancial (UFSM, 2019). Foram submetidos à avaliação do aplicativo desenvolvido 52 OAs, sendo 28 do tipo DOCX e 24 do tipo *ePub*, escolhidos aleatoriamente nos repositórios citados.

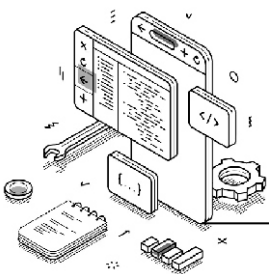
Os testes realizados puderam demonstrar que o aplicativo avaliou corretamente todos os arquivos testados, apenas ressaltando que algumas regras/recomendações, como a presente na avaliação da diretriz sobre o contraste de cores, não puderam ser aplicadas aos OAs no formato *ePub*, devido a problemas nos dados recebidos dos parâmetros, possivelmente em decorrência de problemas na estrutura do arquivo em si.

O aplicativo faz a validação das regras que envolvem textos no formato DOCX e *ePub*. Entretanto, a arquitetura do aplicativo já contempla a possibilidade de inserção de diferentes mídias, deficiências e diretrizes que podem permitir a validação de outros Oas.



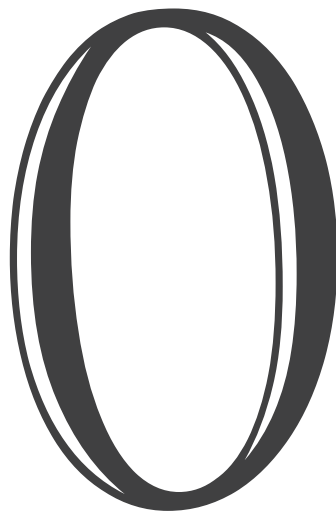
Entre as dificuldades encontradas durante a realização do trabalho, os autores destacam a necessidade de aprendizado das tecnologias envolvidas no desenvolvimento do aplicativo, bem como a existência de alguns problemas na API utilizada para a extração dos parâmetros dos OAs. Quanto às contribuições do trabalho, destacamos a inserção do mesmo nos grupos de pesquisa da UFSM (IATE/UFSM) e da UFSC (Núcleo de Acessibilidade Digital e Tecnologias Assistivas/Universidade Federal de Santa Catarina) (NADTA, 2019). Esta interação entre os grupos de pesquisa permitiu o desenvolvimento de um aplicativo de forma colaborativa, envolvendo diferentes visões e sugestões que aprimoraram a sua implementação.

Como trabalhos futuros os autores elencaram a necessidade de implementação da interface para aquisição do conhecimento, permitindo que sejam inseridas as diretrizes e regras de acessibilidade de OAs em diferentes formatos; a implementação de uma versão web, visando oferecer mais de uma forma de acesso ao aplicativo desenvolvido; o desenvolvimento de uma funcionalidade que permita a gravação do relatório gerado pelo aplicativo na avaliação de um OA, possibilitando visualização posterior; além da atualização na API de Extração de Parâmetros para permitir a extração dos parâmetros de OAs do formato PDF (Portable Document Files).





EXPERIÊNCIAS NO DESENVOLVIMENTO DE SISTEMAS DE RECOMENDAÇÃO

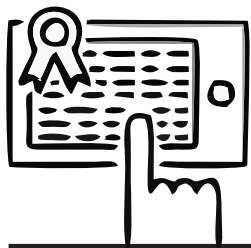
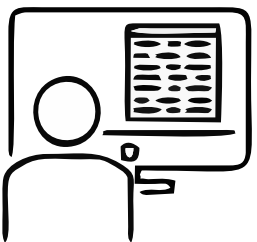
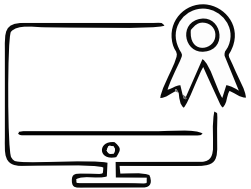


s SRs (Sistemas de recomendação) ou do inglês *Recommender Systems* ou *Recommendation Systems*, são sistemas com poder de analisar e interpretar o comportamento dos usuários de uma plataforma, para fazer recomendações consideráveis e individualizadas de conteúdos novos, como por exemplo, filmes, livros, músicas, lojas, páginas *web*, entre outros, que naturalmente são de interesse do usuário.

O princípio dos SRs fundamenta-se no raciocínio de que “o que é significativo ou interessante para mim da mesma forma pode ser expressivo para outra pessoa com interesse semelhante” (Cheffer & Silveira, 2023; Motta *et al.*, 2011).

Os SRs já são uma realidade em muitos *sites* de compras, onde todos os usuários já vivenciaram uma situação com sugestões relacionadas às informações que acessamos na rede. Os componentes fundamentais de um Sistema de Recomendação são cliente e produto ou, de forma mais genérica, usuários e interesses (o que será recomendado). Um produto pode ser de naturezas distintas, por exemplo, um arquivo, um conteúdo, um objeto, entre outros. Já a recomendação pode ser entendida como uma função de mapeamento de interesses do usuário para aquisição de um ou mais produtos (Cheffer & Silveira, 2023; Motta *et al.*, 2011).

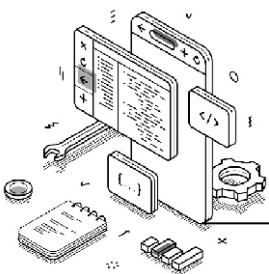
Atualmente, as pessoas têm sido expostas a uma grande quantidade de dados e informações relacionadas às diversas áreas da sociedade. Dessa maneira, o usuário encontra-se em meio a uma imensidão de informações digitais, que em muitos casos não estão organizadas ou caracterizadas. Esse excesso de informação causa uma sobrecarga cognitiva (Lorenzi & Silveira, 2011). Ainda, os processos de busca e recuperação de informação são cada vez mais complexos e o usuário nem sempre tem o conhecimento necessário para criar uma técnica de pesquisa que traga as informações que realmente necessita. Dessa



forma, um dos mecanismos mais usados nos dias atuais para amenizar esse problema é a aplicação dos Sistemas de Recomendação nos diferentes sistemas de informação. Por meio desse recurso, o encontro de dados durante os processos de pesquisa é facilitado, promovendo aos utilizadores indicações de outros produtos e serviços que poderiam estar relacionados aos seus interesses ou necessidades (Alvarez *et al.*, 2016; Cheffer & Silveira, 2023).

Os SRs utilizam diversas técnicas de recomendação que visam a identificar padrões dos usuários como, por exemplo, consumo, pesquisa, comportamento, entre outros para customizar suas buscas. Tais técnicas podem ser definidas em seis grupos principais, listados a seguir (Cheffer & Silveira, 2023; Reis, 2012):

- **Filtragem Colaborativa:** é uma das técnicas mais conhecidas e mais implementadas nos SRs. O conceito passa por sugerir ao usuário itens que outros usuários com preferências similares julgaram interessantes no passado. Nesta técnica, a familiaridade nos interesses de dois usuários é medida com base na afinidade entre o histórico de classificação que os usuários conferiram a cada item;
- **Filtragem Baseada em Conteúdo:** esta técnica tem origem do trabalho de análise na área de filtragem de informações. Os interesses dos usuários são descobertos com base nas propriedades específicas dos itens que este acessou ou classificou. Podemos considerar, como exemplo, o uso das palavras-chave existentes em um texto que o usuário acessou ou classificou ou ainda os nomes dos atores que compõem o elenco de um filme que o usuário adquiriu. A técnica dos métodos baseados em conteúdo aplicam perfis de itens, denominado como grupo de particularidades dos itens, que representam os itens no sistema. Dessa forma, o sistema gera um perfil de conteúdos criando um vetor de pesos relacionados às particularidades dos itens. Tais pesos representam a influência de cada uma dessas particularidades para o usuário e podem ser medidas com base nos vetores relacionados aos itens que o usuário classificou. Com fundamento no perfil do usuário e nos perfis relacionados aos itens o SR utiliza técnicas de correspondência estatísticas ou de aprendizagem computacional para determinar os itens a recomendar;
- **Filtragem Baseada em Conhecimento:** esta técnica busca oferecer recomendações ao usuário baseado no conhecimento em relação aos interesses de um usuário com item em específico. As recomendações são realizadas por meio do estabelecimento de padrões de utilidade, decorrentes com base no conhecimento que se detém das relações de um produto com específico usuário. A recomendação baseada em conhecimento presume uma organização que detenha essas relações e proporcione a sua consulta de maneira a estabelecer a finalidade para o usuário buscando novas recomendações. A compreensão do conhecimento estimado está relacionada somente nas preferências do usuário, assim como o tipo de produto a recomendar;
- **Comunitária:** a técnica de recomendação comunitária é uma das mais conhecidas na atualidade em virtude do seu crescimento e popularidade nas redes sociais. Tal técnica sugere itens aos usuários com relação às



compras dos amigos dos usuários. Dessa maneira, esse modelo de sistema de recomendação acrescenta dados característicos às classificações e recomendações de itens por parte dos usuários que constituem parte de alguma comunidade ou rede, demonstrando relações entre usuários por meio de semelhanças das suas classificações. Novas recomendações são formadas por meio de comparações de características de usuários integrado na mesma rede;

- **Híbridos:** os modelos híbridos são combinações dos vários tipos citados nas categorias anteriores. A compatibilidade de inúmeras abordagens são empregadas constantemente, de forma a preencher os pontos fracos de uma abordagem com os pontos fortes de outras;
- **Demográficos:** a abordagem demográfica caracteriza-se pela recomendação aos usuários com relação às suas propriedades individuais, a começar por um grupo de situações aplicadas aos usuários pelo sistema. As indicações são formadas com base em um mecanismo de escolha que considera o perfil demográfico do usuário.

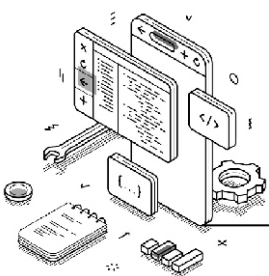
Nas próximas seções apresentamos alguns dos protótipos de Sistemas de Recomendação implementados pelo grupo de pesquisa IATE-UFSM.

3.1 Sistema de Recomendação de Produtos de Pet Shop integrado a uma Loja Virtual (Cheffer & Silveira, 2023)

Esta seção apresenta o desenvolvimento de um Sistema de Recomendação para *Pet Shop*, integrado a uma loja virtual, desenvolvido por Cheffer & Silveira (2023). A motivação para o desenvolvimento deste trabalho surgiu em virtude da inexistência de um sistema de recomendação de produtos de pet shop integrado a uma loja virtual, especialmente na região do Médio Alto Uruguai localizada no estado do Rio Grande do Sul. Dessa forma, verificou-se a necessidade de desenvolver um sistema de recomendação *web* para a loja *Pet Park Hotel*, situada na cidade de Frederico Westphalen - RS, tendo em vista que a mesma oferece serviços diferenciados, tais como o *Day Care* (creche) e hospedagem para *pets*.

Segundo a Associação Brasileira da Indústria de Produtos para Animais de Estimação (ABINPET, 2021), o mercado brasileiro para *pet shops* tem se expandido nos últimos anos. O Brasil ocupa a sexta posição em faturamento no mercado de pets no ranking mundial, sendo uma posição a mais do que no ano de 2020. Conforme esta elevação da posição do país no mercado de pets, as lojas virtuais deste ramo vêm conquistando o mercado. Com a disseminação dos sistemas de compras online ocorre, também, um aumento significativo no número de pessoas que compram online produtos para seus pets (EBIT, 2021).

No Brasil, os animais de estimação são considerados pelos seus donos como integrantes da família ou como filhos. Conforme dados do IBGE (Instituto Brasileiro de Geografia e Estatística) (IBGE, 2021), mais da metade da população tem pelo menos um animal de estimação. Os dados apontam para o crescimento do número de pets em residências. Dados da pesquisa do IPB



(Instituto Pet Brasil) revelam que “...o Brasil encerrou 2021 com 149,6 milhões de animais de estimação, um aumento de 3,7% sobre os 144,3 milhões do ano anterior. Os cães lideram o ranking, com 58,1 milhões de indivíduos. As aves canoras vêm em segundo, com 41 milhões. Os gatos figuram em terceiro lugar, com 27,1 milhões, seguidos de perto pelos peixes (20,8 milhões). E depois vêm os pequenos répteis e mamíferos (2,5 milhões)” (IPB, 2021).

Neste contexto, as compras *online* e a contratação de serviços em diversos segmentos surgem como uma ferramenta imprescindível para o mundo atual, devido aos diversos compromissos diários da sociedade. Ainda existe uma grande dificuldade na busca de produtos específicos para cada espécie de animal, bem como, a dificuldade em encontrar determinados serviços. Dessa maneira, surge a proposta de implementação de um sistema de recomendação de produtos e serviços voltados para pets, integrado a uma loja virtual, que contribuirá para economizar tempo na tomada de decisões, visando praticidade para os donos de pets, pois tornará o processo de aquisição de produtos ou serviços mais simples e ágil.

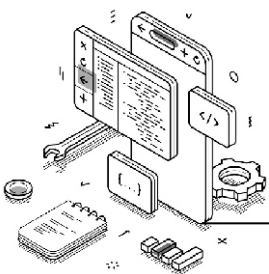
A metodologia de pesquisa utilizada para o desenvolvimento deste trabalho foi a Dissertação-Projeto (Ribeiro & Zabadal, 2010), pois foi desenvolvido um protótipo do SR, integrado à loja virtual.

Para o desenvolvimento deste trabalho foi utilizado, como modelo base a empresa *Pet Park Hotel* (que autorizou a realização do trabalho), que oferece uma linha de serviços voltados para hospedagem, *day care* (creche), vacinas, banho e tosa, além de contar com uma pequena loja de produtos para pets. Para entendimento das regras de negócio e análise de requisitos foram realizadas visitas e reuniões com a sócia proprietária e equipe da referida *pet shop*. Dessa maneira, foi possível compreender as necessidades da empresa para desenvolver o protótipo.

Para o desenvolvimento do protótipo foram empregadas algumas tecnologias e ferramentas, tais como o *framework Bootstrap*, a linguagem de programação PHP com o *framework Laravel* e a linguagem de programação *JavaScript*.

Para o levantamento dos requisitos os autores realizaram uma reunião com a sócia proprietária do *Pet Park Hotel*, para entender as necessidades do sistema desenvolvido. A partir da reunião foram definidos:

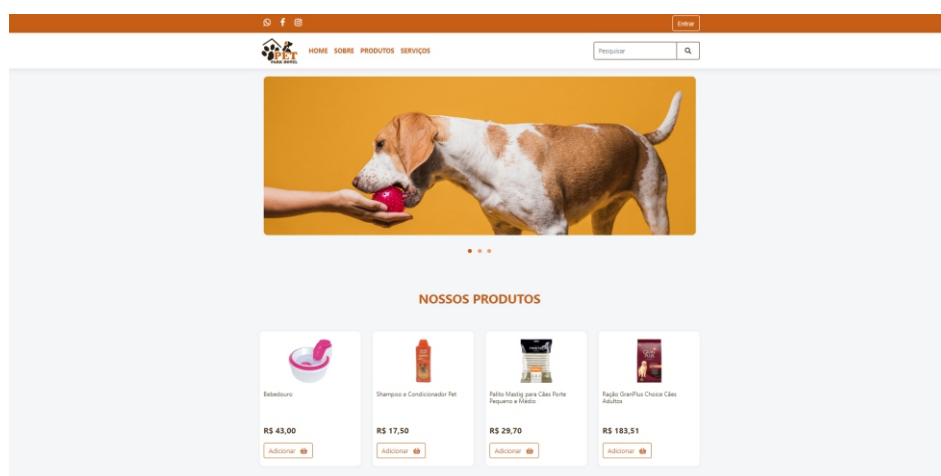
- Desenvolvimento do Banco de Dados: foi definida a utilização do Sistema Gerenciador de Bancos de Dados *MySQL*, para gerenciar os cadastros e consultas de acordo com a proposta do trabalho;
- Painel de administração do SR: definiu-se o desenvolvimento de um painel de administração, no qual o usuário administrador pode cadastrar e gerenciar as informações dos produtos e serviços, gerenciar pedidos e usuários;
- Loja virtual: desenvolvimento de uma loja virtual utilizando o *framework Laravel*. Neste sistema é possível cadastrar novos usuários e gerenciar informações de cadastro, visualizar e escolher produtos para efetuar um



pedido, recomendação de produtos, acompanhamento de pedidos e agendamento de serviços.

Como pode ser observado na Figura 17, as funcionalidades essenciais como acesso aos produtos, serviços e formulário para pesquisa estão presentes no menu principal do sistema.

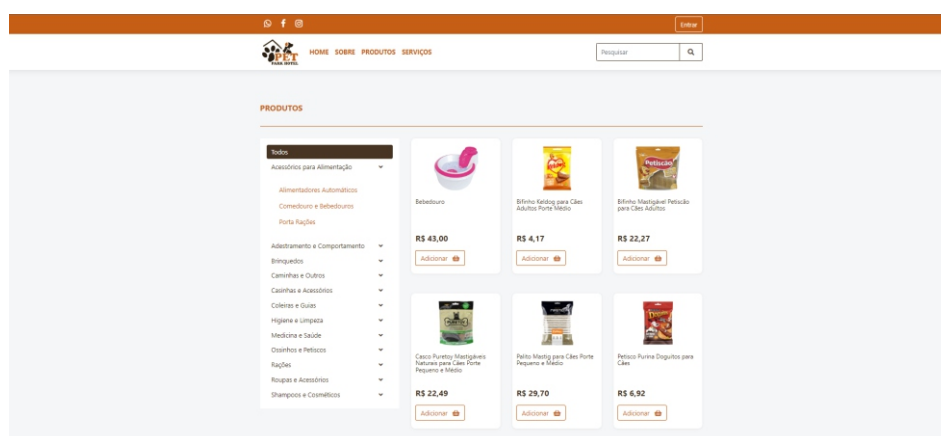
Figura 17: Tela Principal do SR.



Fonte: Cheffer & Silveira, 2023

Como pode ser visto na Figura 18, todos os produtos cadastrados no sistema que estão ativos são listados nessa tela. No lado esquerdo desta página, é exibida a lista de categorias e subcategorias que permite a filtragem dos produtos, facilitando a busca pelo produto desejado.

Figura 18: Tela de Produtos.



Fonte: Cheffer & Silveira, 2023

Após estar *logado* no sistema, o usuário poderá adicionar produtos ao seu

Para o desenvolvimento do SR foi aplicada a técnica de recomendação baseada em conteúdo (Lorenzi & Silveira, 2011) para considerar a experiência do usuário alvo. Quando um novo cliente se cadastra no sistema, é realizada a coleta, de forma explícita, sobre o interesse em determinados produtos. Desse modo, é possível desenhar um perfil inicial do usuário e a partir disso, o sistema já consegue sugerir produtos que podem interessar o cliente.

Para gerar as recomendações, são realizadas coletas de forma explícita por meio de formulário de cadastro (Figura 19), onde o usuário marca as categorias que interessam a ele (*gerenciar preferências*). Também é realizada a coleta de forma implícita, no momento em que o usuário adiciona um produto ao pedido.



HOME

SOBRE

PRODUTOS

SERVIÇOS

Inscrever-se

CADASTRO

Nome *

CNP *

E-mail *

Telefone *

Estado *

Selecione o estado

Cidade *

CEP *

Bairro *

Assinatura *

Nome *

Senha *

Confirme a senha *

GERENCIAR PREFERÊNCIAS

☐ Acessórios para Alimentação

☐ Adesivos e Comportamento

☐ Brinquedos

☐ Cadeiras e Cadeiras

☐ Cadeiras e Acessórios

☐ Cadeiras e Cadeiras

☐ Higiene e Limpeza

☐ Medicina e Saúde

☐ Orelhas e Orelhas

☐ Raças

☐ Raças e Acessórios

☐ Shampoos e Cosméticos

Calcular

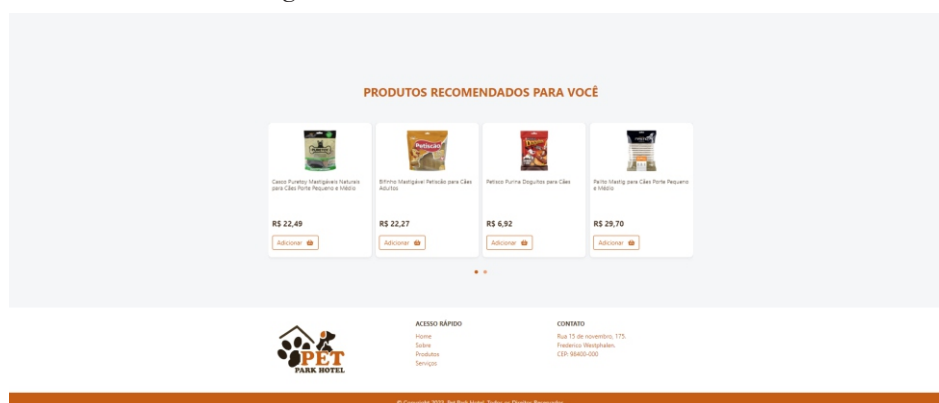
Para criar as recomendações, foram utilizados dados já cadastrados pelo administrador do sistema, por meio do painel de administração. No formulário de cadastro de produto é definida a categoria que o produto pertence, e a mesma é utilizada para indicar similaridades entre os produtos, sendo uma das características principais da recomendação dos produtos.



Por meio das informações dos produtos cadastrados no banco de dados, é possível capturar, de forma explícita, as preferências do usuário, no momento em que ele cria seu cadastro no sistema. Desse modo, é possível definir um perfil inicial de interesses do usuário para gerar recomendações já no primeiro contato com o sistema.

Após a efetivação do cadastro, ao acessar a conta, o usuário poderá visualizar os produtos recomendados em algumas telas como, por exemplo, no final da página *Home*, conforme mostra a Figura 20.

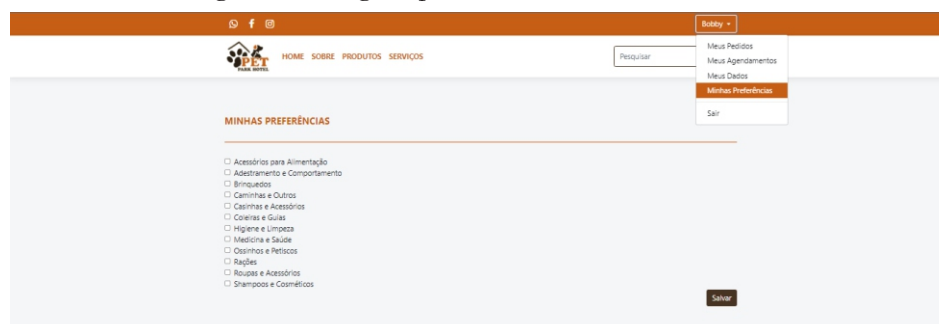
Figura 20: Produtos Recomendados.



Fonte: Cheffer & Silveira, 2023

Caso as sugestões fornecidas pelo sistema não estiverem mais alinhadas com as preferências do usuário, ele tem a opção de personalizá-las em seu perfil. Isso pode ser feito acessando o menu "Minhas Preferências" localizado na barra superior direita, como pode ser visto na Figura 21.

Figura 21: Página para Gerenciar Preferências.



Fonte: Cheffer & Silveira, 2023

Durante o desenvolvimento do sistema, os desenvolvedores realizaram testes para verificar se as funcionalidades implementadas estavam realizando as

operações previstas de forma adequada. Ao final da implementação, foi realizada a etapa de testes e validação junto com a sócia proprietária da empresa *Pet Park Hotel*, localizada na cidade de Frederico Westphalen - RS. Esta validação é conhecida como avaliação por um especialista, ou seja, um usuário que conheça o domínio da aplicação (Maurício Neto, 2005). Dessa forma, a sócia proprietária acompanhou o funcionamento do sistema.

A validação do sistema com a especialista (sócia proprietária do *Pet Park Hotel*) teve duração de cerca de cinquenta minutos. Os desenvolvedores apresentaram o protótipo em funcionamento, explicando as funcionalidades desenvolvidas. A especialista considerou o protótipo adequado, afirmando que o layout do sistema é limpo, intuitivo e de fácil manuseio. Como melhorias futuras, a especialista propôs que fossem adicionados alguns critérios para os serviços de hospedagem e creche, tais como carteira de vacinação do pet em dia, e materiais essenciais do pet, tais como cama, cobertor e comida. Outra informação necessária para a seção de serviços é a opção de transporte do pet, onde o cliente poderia optar pela forma que melhor atendesse suas necessidades.

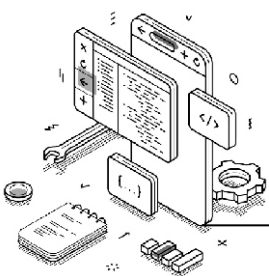
Com o objetivo de validar a usabilidade do protótipo de sistema desenvolvido, elaborou-se um instrumento de pesquisa, na forma de um questionário para aplicar aos usuários possíveis clientes após o primeiro contato com o sistema de recomendação. A aplicação do questionário foi realizada de duas formas, por meio vídeo chamada utilizando a ferramenta de acesso remoto *AnyDesk*, e de forma presencial. No total, dez pessoas responderam ao questionário e tiveram o primeiro contato com o sistema.

Os resultados positivos, obtidos por meio da validação com a especialista da área e, também, com possíveis clientes, mostram que o protótipo implementado está adequado e oferece as funcionalidades de acordo com as expectativas dos usuários.

Para trabalhos futuros, o SR e a loja virtual podem ser aplicados, efetivamente, para fins comerciais. Para isso, é necessário hospedar o SR na *web* e realizar a integração com uma API de pagamento e desenvolvimento de um módulo para controle financeiro das vendas e prestação de serviços.

3.2 Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom (Colli & Silveira, 2023)

A partir do início do ano de 2020, devido à situação pandêmica vivida pela disseminação do vírus COVID-19 (também conhecido como Coronavírus), foi necessária a criação de sistemas de contenção e isolamento, afetando a sociedade como um todo. Estabelecimentos de Ensino – creches, escolas, universidades – tiveram suas atividades escolares presenciais suspensas, o que atingiu milhões de estudantes em todo o mundo. Houve, assim, um grande impacto social, que levou um tempo para ser absorvido pelo público e também pelos gestores educacionais, que ficaram atônitos e com isso uma reação



demorou um pouco a ocorrer (Pasini, Carvalho e Almeida, 2020). Por ter sido um impacto repentino, houve uma alta demanda imediata de métodos *online* que pudessem contribuir para o ensino remoto.

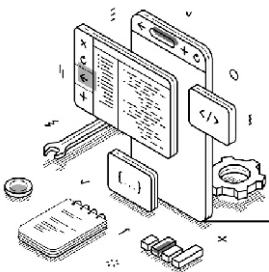
A motivação para o desenvolvimento do trabalho aqui apresentado, desenvolvido por Colli & Silveira (2023), surgiu mediante à análise do cenário mundial atual e a constatação da necessidade da intensificação na utilização de diferentes TDICs na Educação, especialmente devido ao isolamento social, decorrente da Pandemia de COVID-19 que levou à necessidade de adaptação a novas maneiras de ensinar e aprender, de forma remota, permitindo o acesso rápido a diferentes conteúdos de temas diversos. Tendo em vista esta necessidade, destaca-se a utilização de OAs como forma de auxílio para a modalidade de ensino remoto, híbrido e, também, na modalidade de EaD. Em todos os níveis de ensino, os diferentes meios fornecidos pelos OAs trazem facilidade a um ensino que muitas vezes é considerado um desafio, devido aos diferentes perfis e estilo cognitivos do público alvo.

Os OAs tendem a ser utilizados para apoiar os processos, seja de ensino e/ou de aprendizagem, nas modalidades presencial, híbrida e/ou a distância. É primordial saber onde encontrar os objetos adequados à proposta pedagógica utilizada pelo professor. Atualmente existem repositórios de objetos de aprendizagem de diversas áreas, tanto no Brasil como no exterior. Com isso, o ponto a se destacar é a quantidade de informação que está disponível para ser acessada na *web*, o que pode causar uma sobrecarga cognitiva sobre o usuário final (Freitas, Silveira & Bertolini, 2019), já que existe uma liberdade na disposição destes materiais, muitas vezes sem supervisão, e de forma desorganizada. A partir disto, surgiu a proposta de implementar um SR, para filtrar e indicar aos alunos os OAs mais adequados, considerando os seus estilos cognitivos, baseando-se na Taxonomia de *Bloom*, para definir a proposta pedagógica dos OAs indicados.

Neste contexto, o principal objetivo deste trabalho foi o de desenvolver um protótipo de SR para recomendar Objetos de Aprendizagem, baseado nos estilos cognitivos dos estudantes e na Taxonomia de *Bloom* (Colli & Silveira, 2023).

Os OAs - ou *Learning Objects*, também conhecidos como Objetos Educacionais - ou *Educational Objects*, têm, como principal objetivo, o de prestar auxílio a toda e qualquer modalidade de ensino, com ferramentas voltadas à tecnologia. Seu conceito vem sofrendo alterações, mas pode-se destacar que são unidades formadas por um conteúdo didático, estes que podem ser vídeos, animações, textos, gravações, imagens, entre outros dos mais diversos, com a missão de agregar conhecimento (Parreira, Falkembach & Silveira, 2018).

A solução implementada neste trabalho envolveu o desenvolvimento de um protótipo de SR de Objetos de Aprendizagem, baseado em estilos cognitivos e na Taxonomia de *Bloom*. Este protótipo traz um cadastro e acesso para usuários e professores, visando diferenciar sua utilização no SR. O professor, a partir do seu *login*, pode cadastrar OAs, categorizando-os mediante os estilos cognitivos e classificação de acordo com a Taxonomia de *Bloom*. Os usuários (alunos), mediante cadastro e preenchimento do instrumento para



compreender o seu estilo cognitivo (modelo de aluno), recebem recomendações de acordo com seu estilo e, também, de acordo com a Taxonomia de *Bloom* em que o OA se encontra.

A metodologia de pesquisa utilizada para o desenvolvimento deste trabalho foi a Dissertação-Projeto (Ribeiro & Zabadal, 2010), pois foi desenvolvido um protótipo do SR.

Com o intuito de validar este sistema, foi solicitado o acompanhamento de professores e alunos do curso de bacharelado em Sistemas de Informação, da Universidade Federal de Santa Maria (UFSM), campus Frederico Westphalen/RS. Esses também participaram da fase de testes durante o desenvolvimento, com práticas de utilização, sugerindo melhorias para facilitar a navegação e uso do sistema, também sugerindo funcionalidades adicionais, caso necessário. Além disso, foram realizados testes com alunos e professores do município de Sarandi/RS e, posteriormente, aplicamos um formulário para validação do protótipo apresentado.

O SR desenvolvido compreende os alunos em seis (6) estilos cognitivos, a partir das respostas dadas ao questionário proposto por Bariani (1998) e implementado computacionalmente por Geller (2004). O aluno, após fazer o cadastro no SR, responde ao questionário referente aos estilos cognitivos. O SR armazena os dados dos alunos, bem como o estilo cognitivo preferencial (dominante). Os estilos cognitivos são apresentados no Quadro 2.

Quadro 2 - Dimensões dos estilos cognitivos

Convergente	Divergente
Lógico, disciplinado, conservador	Criativo, imaginativo
Holista	Serialista
Trabalha no contexto global, com hipóteses complexas	Trabalha com tópicos separados e com sequência lógica
Impulsivo	Reflexivo
Detém-se pouco em ponderação e organização prévia	Possui um pensamento organizado

Fonte: Adaptado de Bariani (1998) e Geller (2004)

Após a identificação do estilo cognitivo dominante dos alunos, a partir dos metadados dos Objetos de Aprendizagem cadastrados pelos professores, o algoritmo de recomendação define os OAs mais adequados, unindo as características de aprendizagem dos estilos cognitivos às atividades propostas na Taxonomia de *Bloom*. Para todos os estilos cognitivos utiliza-se a base da pirâmide (conhecimento). A Figura 22 apresenta a taxonomia de Bloom.

A Taxonomia de *Bloom* é uma classificação dos domínios da aprendizagem, a partir da listagem das competências e dos processos envolvidos nas atividades educacionais, e com isso estabelece critérios avaliativos. Serve, então, para definir os objetivos da aprendizagem e planejar aulas respeitando a hierarquia dos mesmos (SAE, 2022).

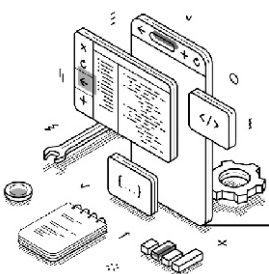
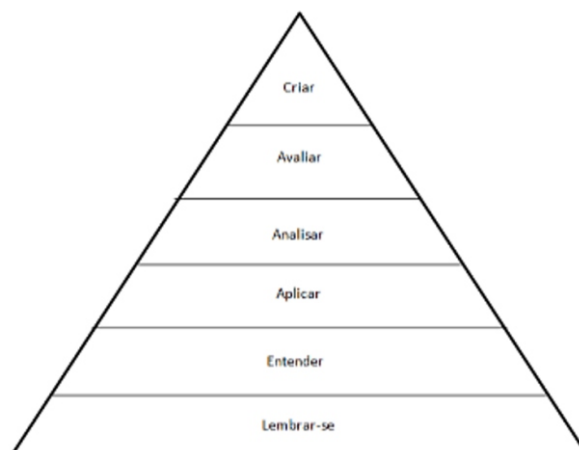


Figura 22: Taxonomia de Bloom.



Fonte: Adaptada de Bergamnn (2018)

Muitas vezes ela é representada em formato de pirâmide. A ideia é que a base desta pirâmide envolva as atividades mais simples, que poderiam ser realizadas em casa, como no dever de casa, por exemplo, e à medida que se avance os níveis na pirâmide, o nível de complexidade e dificuldade aumente em contrapartida (Silveira, Parreira & Bigolin, 2019).

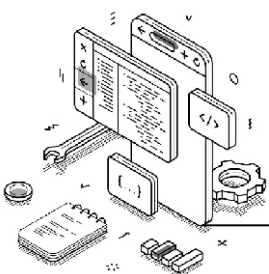
O Quadro 3 apresenta os níveis da pirâmide da Taxonomia de *Bloom* que são aplicados na recomendação dos OAs de acordo com os estilos cognitivos.

Quadro 3 - Dimensões dos estilos cognitivos de acordo com a Taxonomia de *Bloom*

Convergente	Divergente
Lembrar-se + Aplicar	Lembrar-se + Compreender
Holista	Serialista
Lembrar-se	Lembrar-se + Analisar
Impulsivo	Reflexivo
Lembrar-se + Compreender + Aplicar	Lembrar-se + Avaliar + Criar

Fonte: Colli & Silveira (2023)

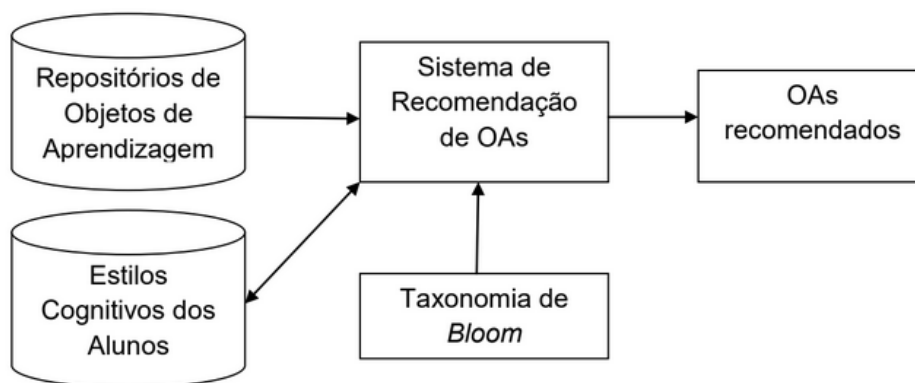
Cabe destacar que a compreensão dos estilos cognitivos é essencial para aprimorar os processos de ensino e de aprendizagem, visando a potencializar a aprendizagem, construindo um modelo de aluno que possa ser aplicado, especialmente, em sistemas tecnológicos. Alguns pesquisadores afirmam que estudantes obtêm melhor desenvolvimento quando o estilo de ensino equivale ao seu estilo cognitivo. Já outros argumentam que a variação nos métodos de ensino baseado nos diversos estilos cognitivos tende a surtir um efeito mais significativo. Mesmo com esta divergência, pode-se salientar sua importância na viabilização de práticas educacionais mais eficazes, uma vez que podem



influenciar na forma que os professores ensinam, que os alunos aprendem, e como em um ambiente eles interagem (Geller, 2004).

A Figura 23 apresenta uma arquitetura de alto nível do Sistema de Recomendação implementado.

Figura 23: Arquitetura de Alto Nível do SR.

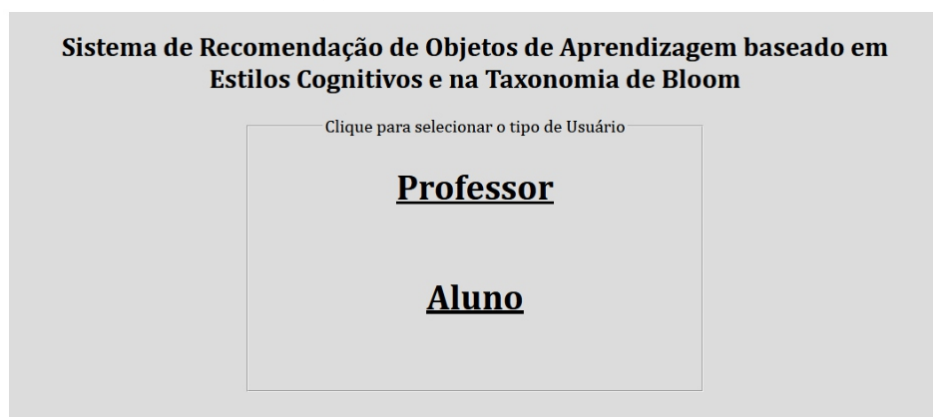


Fonte: Colli & Silveira (2023)

No desenvolvimento do SR foram empregadas as seguintes tecnologias de *software*: *IDE (Integrated Development Enviroment)* de desenvolvimento *NetBeans*; Sistema Gerenciador de Banco de Dados *MySQL*; Servidor *Apache* para execução do SR; linguagem de programação *PHP*; linguagem de marcação *HTML (HyperText Markup Language)*; folhas de estilo *CSS (Cascade Style Sheets)*. Definimos as tecnologias *PHP*, *HTML*, *CSS* e *MySQL* para a implementação por questões de domínio de linguagem e por se tratar de melhor adequação ao objetivo proposto, voltados para a *web*.

A Figura 24 traz a tela inicial do SR implementado, sendo definidos acessos diferentes para os atores Professor e Aluno, visando uma interface mais intuitiva para o usuário final, além de uma interface para o Administrador do sistema.

Figura 24: Tela Inicial do SR.



Fonte: Colli & Silveira (2023)

Após o usuário realizar seu acesso, será apresentado a ele um menu, apresentado na Figura 25, contendo as opções de funcionalidades para os professores, que são: Cadastrar Objeto de Aprendizagem, Gerenciar meus Objetos de Aprendizagem e Gerenciar meu Cadastro de Usuário. Os professores são os responsáveis pelo cadastro de OAs no SR, para que os mesmos possam, posteriormente, serem recomendados aos alunos de acordo com seus estilos cognitivos.

Figura 25: Tela do SR com as opções para o Perfil Professor.

The screenshot shows a web interface titled "Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom". Below the title is a box labeled "Menu - Professor" containing three bullet points: "Cadastrar Objeto de Aprendizagem", "Gerenciar meus Objetos de Aprendizagem", and "Gerenciar meu Cadastro de Usuário". At the bottom of the box is a "Sair" button.

Fonte: Colli & Silveira (2023)

Clicando em *Cadastrar Objeto de Aprendizagem* (Figura 25), o professor será redirecionado à tela da Figura 26, onde, como aparece na imagem, foi simulado um cadastro com os dados preenchidos.

Figura 26: Cadastro de Objetos de Aprendizagem.

The screenshot shows a web interface titled "Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom". Below the title is a box labeled "Cadastro de Objeto de Aprendizagem" containing several form fields: "ID do OA:" with the value "1218", "Descrição:" with the text "Artigo sobre a 'Fórmula Quadrática'", "Área:" with the value "Matemática", "Nível da Taxonomia:" with a dropdown menu showing "Entender", and "Link:" with the URL "https://pt.wikipedia.org/wiki/F%C3%B3rmula_quadr%C3%A1tica". At the bottom of the box are two buttons: "Realizar Cadastro" and "Voltar".

Fonte: Colli & Silveira (2023)

Destaca-se o preenchimento da área em que o OA se enquadra e o Nível da Taxonomia de *Bloom*, este último que se encontra dentro de uma caixa de seleção, a qual lista os dados que constam no banco na tabela taxonomia. Vale ressaltar que o ID é validado via algoritmo, para não haver conflito de dados iguais no SR.

No caso do perfil de *Aluno*, ao acessar o SR pela primeira vez, é preciso responder ao questionário que irá compreendê-lo nos estilos cognitivos. O aluno será direcionado à tela com a Escala de Avaliação de Estilos Cognitivos, apresentada na Figura 27, onde deverá responder a cada uma das 18 questões em um grau de “Discordo Totalmente” até “Concordo Totalmente”, os quais contarão uma pontuação (de 1 a 5, em ordem das respostas) no algoritmo, para definir o estilo cognitivo dominante do Aluno, conforme critérios definidos por Bariani (1998).

Pode-se destacar que o SR faz a verificação das questões antes do envio, conferindo se todas estão marcadas, e caso houver alguma sem resposta apresenta um erro ao usuário destacando qual é a pergunta.

Figura 27: Escala de Avaliação dos Estilos Cognitivos.

Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom

Cadastro - Aluno

[Voltar à tela de Login](#)

Primeiramente, preencha a **Escala de Avaliação de Estilos Cognitivos**.
Pense no que acontece com maior frequência e não deixe nenhum item sem resposta.

1. Eu considero difícil criar algo original.

- ☐ Discordo Totalmente
- ☐ Discordo
- ☐ Indeciso
- ☐ Concordo
- ☐ Concordo Totalmente

2. Em muitas situações, eu não sou uma pessoa atenta, porque sou apressado.

- ☐ Discordo Totalmente
- ☐ Discordo
- ☐ Indeciso
- ☐ Concordo
- ☐ Concordo Totalmente

Fonte: Colli & Silveira (2023)

Após realizar o cadastro e responder ao questionário (Figura 27) os alunos podem utilizar a opção *Receber Recomendações de Objetos de Aprendizagem*, sendo direcionados à tela da Figura 28, onde é apresentado o seu estilo Cognitivo dominante. A partir disso, receberá recomendações de OAs de acordo com estilo cognitivo e os diferentes níveis da taxonomia de *Bloom*, os quais contêm um *link* clicável que abre em uma nova aba do navegador o conteúdo em questão.

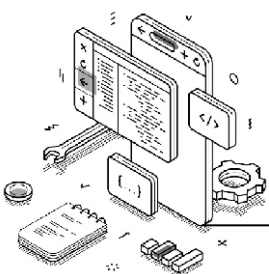
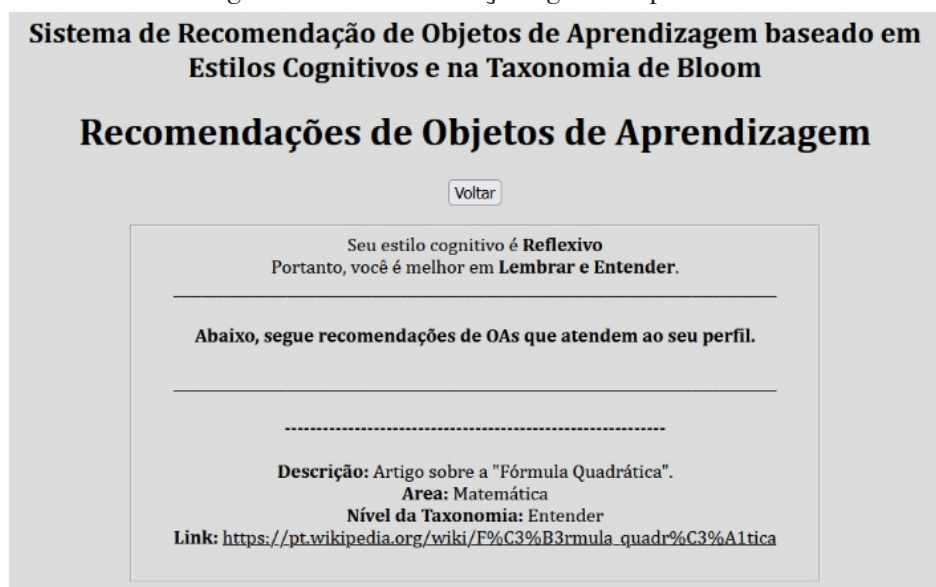


Figura 28: Recomendações geradas pelo SR.

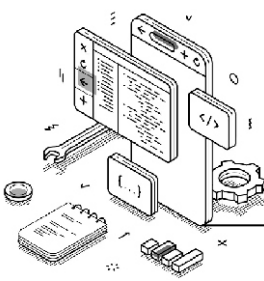


Fonte: Colli & Silveira (2023)

Durante o desenvolvimento do protótipo foram realizados vários testes, a fim de verificar se todas as suas funcionalidades estavam sendo executadas corretamente, bem como verificar se o sistema atende e auxilia de forma adequada professores e alunos. Após a finalização da fase de implementação, o sistema foi apresentado a um grupo de 10 voluntários, de forma remota, onde 6 foram identificados como alunos e 4 identificados como professores, do município de Sarandi/RS, a fim de verificar se as funcionalidades propostas estavam operando corretamente. A seguir, foi lhes entregue um questionário para validação do protótipo. Destacamos que os resultados da avaliação, em geral, foram positivos, evidenciando a aprovação unânime no aspecto de recomendações geradas, 80% no aspecto de navegação e 90% no aspecto referente à intuitividade. Os resultados positivos fazem crer que o protótipo de SR desenvolvido atende os objetivos elencados pelo trabalho.

Existem melhorias a serem feitas no algoritmo de recomendação para trabalhos futuros, onde podemos destacar a possibilidade de um usuário ser compreendido em mais de um estilo cognitivo, e com isso aumentar a gama de conteúdo que este poderá receber nas recomendações de OAs. E ao que se trata do sistema em geral, dispor de um mecanismo de busca para os OAs, onde o usuário poderá filtrar o conteúdo de suas recomendações, além de trazer melhorias na interface com a adição de um *Front-end* mais elaborado.

Acreditamos que o protótipo de Sistema de Recomendação implementado atende seus objetivos, tendo sido testado e validado por usuários reais, além de apresentar uma ferramenta extremamente útil no que engloba a compreensão dos estilos cognitivos dos alunos, e sua aplicação na Taxonomia de *Bloom*, gerando recomendações para aprimorar e auxiliar o aprendizado dos alunos.



3.3 Sistema de Recomendação Mobile: um estudo de caso para delivery (Garbin & Silveira, 2017)

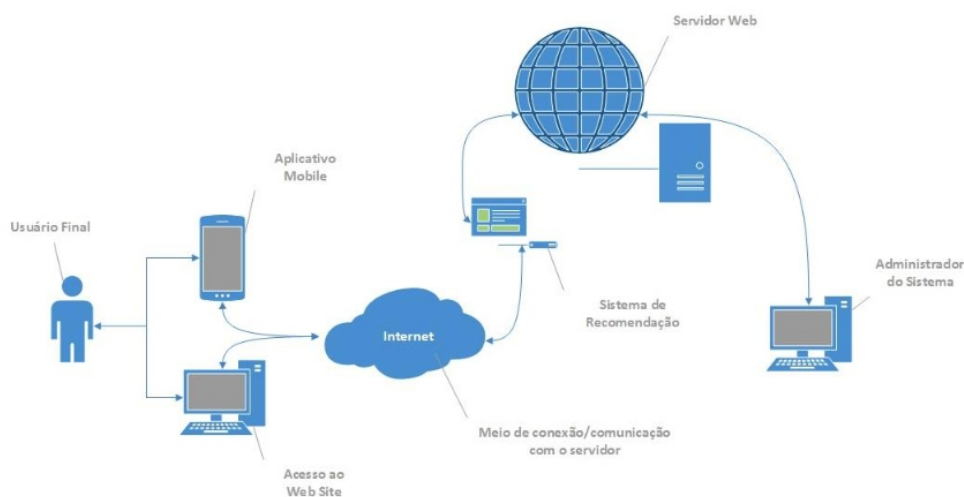
O sistema apresentado nesta seção, de autoria de Garbin & Silveira (2017), envolve o desenvolvimento de um SR, utilizando recursos de computação móvel (plataforma *mobile*). Um fator importante para o desenvolvimento deste trabalho é o uso de software livre, tais como o sistema operacional *Android*, os *frameworks* *Phonegap*, *CodeIgniter 2* e *Jquery*, além da linguagem de programação PHP e os bancos de dados *MySQL* e *SQLite* juntamente com a linguagem de marcação HTML, CSS e a linguagem de programação *JavaScript*.

Portanto, ao agregar as tecnologias propostas neste trabalho, os autores desenvolveram uma aplicação capaz de agregar um diferencial na utilização do serviço de entrega na área de alimentação – que é a área compreendida no estudo de caso, principalmente por estabelecimentos como pizzarias e lanchonetes, sobretudo os situados em nosso mercado regional. Sendo assim, acreditamos que esse diferencial possibilitará ao estabelecimento, além da organização dos pedidos, fidelização dos clientes atuais e também a busca por novos clientes em potencial, ter seu atendimento e relacionamento, junto ao cliente, melhorado, contribuindo assim para um possível acréscimo em suas vendas.

O diferencial de uma empresa, tanto para manter como para atrair novos clientes, está principalmente ligado ao atendimento e conforto do cliente, além da organização da mesma. Um dos pontos importantes da solução implementada está no uso de um SR, o qual auxilia usuários a fazer escolhas de acordo com seu interesse, perfil ou de indicações de outros usuários.

O desenvolvimento da aplicação aqui apresentada foi dividido em quatro etapas. Na primeira etapa foi realizado um estudo sobre algumas tecnologias que envolvem o desenvolvimento de aplicações *web* e também de aplicações para dispositivos móveis juntamente com sistemas de recomendação. A Figura 29 apresenta a arquitetura do sistema de recomendação implementado.

Figura 29: Arquitetura do SR.



Fonte: Garbin & Silveira (2017)

A segunda etapa foi composta pelo levantamento de requisitos, tanto funcionais, como não-funcionais. A terceira etapa envolveu a modelagem do sistema. Esta modelagem foi composta por diagramas e casos de uso que fazem parte da aplicação implementada. A quarta etapa envolveu a validação do sistema. Esta validação foi realizada por meio do acompanhamento por um especialista da área e, posteriormente, a implantação do sistema em uma empresa local, como estudo de caso, no ramo de pizzarias.

Após o levantamento de informações básicas sobre o funcionamento de sistemas do tipo *delivery*, os autores entraram em contato com um comerciante local, sendo exposto, de forma verbal, o funcionamento do sistema proposto, verificando se o mesmo estaria de acordo com os procedimentos com os quais seu comércio trabalha atualmente. Com esta entrevista informal foi possível obter alguns levantamentos de requisitos para a implementação do sistema, dentre eles foi possível definir:

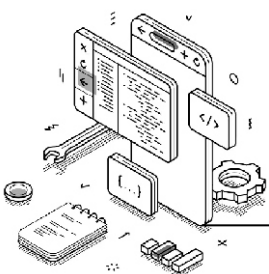
- Implementação do Banco de Dados: foi utilizado o banco de dados *SQLite* para o aplicativo *mobile*, sendo que o banco de dados principal da aplicação é o *MySQL*. O banco de dados principal servirá para gerenciar os cadastros, consultas e relatórios de acordo com a proposta do trabalho;
- Administração do Sistema: foi desenvolvido um módulo de administração do sistema. Neste módulo é possível cadastrar e gerenciar informações dos produtos, gerenciar pedidos, gerenciar usuários, visualizar relatórios por meio de uma interface *web* para *desktop*;
- Aplicativo *Mobile*: foi realizada a implementação de um aplicativo *mobile* utilizando o *framework Phonegap*. Neste aplicativo pode ser feito o cadastro de um novo usuário, configuração de perfil, escolha de produtos para efetuar um pedido, recomendações de produtos e acompanhamento do pedido;
- *Web Site*: criação de um *web site* utilizando *frameworks* para o auxílio no desenvolvimento da aplicação. Este *web site* visa o atendimento de usuários que não possuem um dispositivo móvel.

A implementação da interface e criação do *layout web* foi realizada por meio da utilização do *framework front-end*. Com ele é possível a criação de *layout* responsável que, segundo Silva (2014), deve ser entendido como o design capaz de “responder” às características do dispositivo ao qual é servido.

Responder, nesse contexto, tem o sentido de movimentar-se expandindo e contraindo os componentes da interface do sistema, de acordo com as dimensões e características do dispositivo que o usuário estiver utilizando.

Para os testes de interface e funcionamento em dispositivo móvel, foi utilizado o aplicativo *Phonegap Developer App* instalado em um *smartphone Motorola Razr HD*, este aplicativo está disponível para os sistemas operacionais *Android*, *iOS* e *Windows Phone*.

Para a geração das recomendações foram utilizados dados já cadastrados pelo administrador do sistema por meio do módulo de administração. Esses dados



são os itens que indicam a categoria, subcategoria e o produto, que neste caso são as *pizzas*. A categoria e subcategoria são características utilizadas para indicar a similaridade entre os produtos, para auxiliar no processo de recomendação. Com estes dados cadastrados na base de dados é possível efetuar a coleta de informações do usuário de forma explícita na hora de seu cadastro. Dessa forma é possível traçar o perfil inicial e gerar recomendações para o mesmo.

Quando um novo usuário efetua seu cadastro no sistema, é solicitado que o mesmo entre com suas preferências de recomendação, conforme mostra a Figura 30.

Figura 30: Preferências dos Usuários.

The screenshot shows a web form for user registration. At the top, there's a navigation bar with links like HOME, QUEM SOMOS, COMO FUNCIONA, FAÇA SEU PEDIDO, and CONTATO. The main heading is 'Cadastro de Usuário > Novo'. The form fields include:

- Nome (First Name)
- Sobrenome (Last Name)
- E-mail
- Senha (Password)
- CPF
- Data de Nasc.
- Telefone Fixo
- Telefone Celular

 Below these is a 'Preferências' section with a grid of radio buttons for selecting favorite pizza types:

- Pizza Doce: Doce, Salgada, Portuguesa
- Pizza Salgada: Salgada, Quente, Fritada
- Pizza Quente: Quente, Salgada, Fritada
- Pizza Fritada: Fritada, Salgada, Quente
- Pizza Especial: Especial, Salgada, Quente
- Pizza Margherita: Margherita, Salgada, Quente

 At the bottom of the form are 'Cancelar' and 'Cadastrar Usuário' buttons. Below the form is a green banner with icons and text:

- TELE-ENTREGA GRÁTIS (CENTRO E BAIRROS)
- É RÁPIDO É FÁCIL! FAÇA SEU PEDIDO ON-LINE!
- INFORMAÇÕES: Quem Somos, Como funciona?, Contato, Horários de Entrega, etc.
- ENTRE EM CONTATO: Endereço, Telefone, E-mail, etc.
- ATENDIMENTO: Horário de Atendimento, etc.

Fonte: Garbin & Silveira (2017)

Para ativar e visualizar as recomendações para o usuário é necessário clicar em “Desejo Recomendações” no sistema *web* e “Recomendações” no aplicativo *mobile*. Na Figura 31 são apresentadas as recomendações na interface *mobile*, onde é possível visualizar duas listas de itens que foram recomendados e que possuem relação com as preferências do usuário *logado* no sistema.

Conforme mostra a Figura 31, a primeira lista - que possui o título de “Recomendamos para você” - é baseada nas similaridades definidas pelo especialista do domínio (administrador do sistema). Na segunda lista - “Recomendações baseadas nas avaliações de outros clientes” ou “Recomendações de outros usuários” são apresentadas as recomendações baseadas nas avaliações dos clientes que deram notas aos itens de pedidos concluídos.

Caso as recomendações oferecidas pelo sistema não sejam mais compatíveis com as preferências do usuário, o mesmo poderá editá-las em seu perfil clicando no menu “Preferências de Recomendação”.

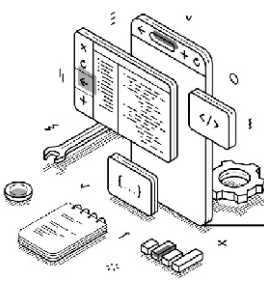
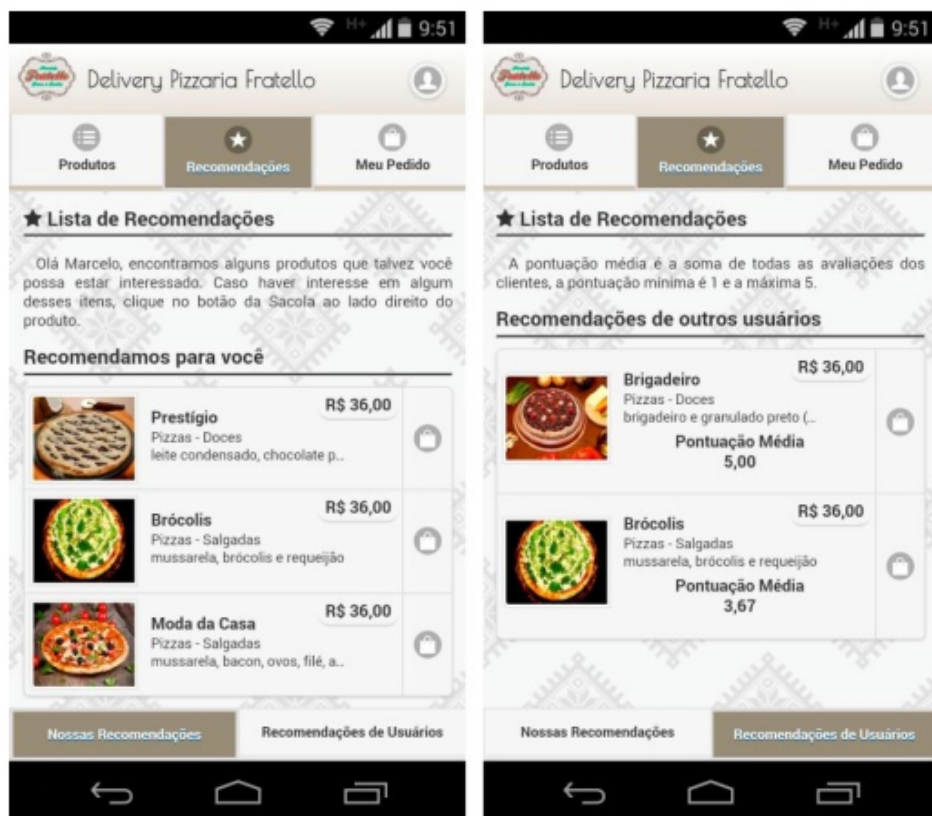


Figura 31: Recomendações geradas pelo SR.

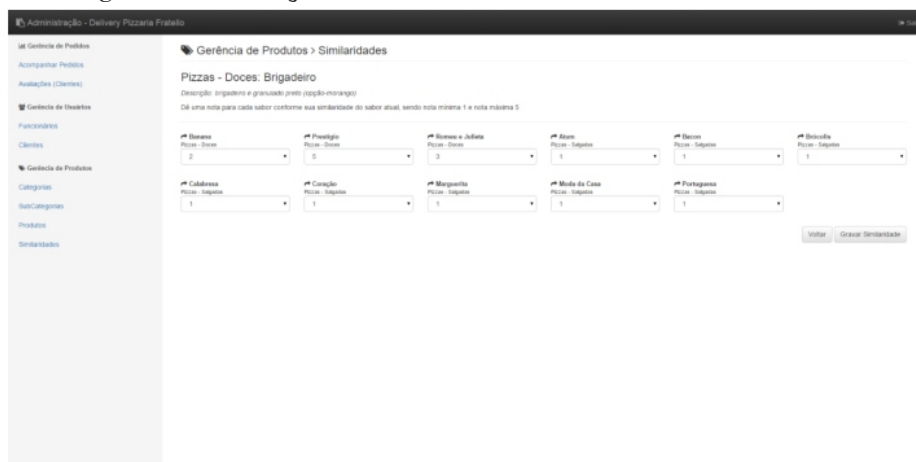


Fonte: Garbin & Silveira (2017)

Caso as recomendações oferecidas pelo sistema não sejam mais compatíveis com as preferências do usuário, o mesmo poderá editá-las em seu perfil clicando no menu “Preferências de Recomendação”.

Os tipos de filtragem que foram empregados neste sistema são a baseada em conteúdo e a colaborativa. Na filtragem baseada em conteúdo, são utilizadas as similaridades cadastradas pelo especialista do domínio.

Figura 32: Definição das Similaridades entre os Produtos do SR.



Fonte: Garbin & Silveira (2017)

Na Figura 32, podemos ver a tela em que se define o quanto similar os itens são com o item atual selecionado, a nota para a similaridade é de 1 para pouco similar e máximo 5 para muito similar ao item selecionado.

Na filtragem colaborativa, foi realizado o cálculo da média das avaliações dos usuários para os produtos e são mostrados os produtos que foram melhor avaliados de acordo com a preferência do usuário *logado* no sistema. Os autores pretendem, futuramente, buscar as avaliações apenas dos clientes com gostos similares. A Figura 33 apresenta a interface para que o usuário avalie os pedidos no aplicativo móvel, avaliação esta que serve como entrada para o cálculo da média referida.

A avaliação de itens proposta neste trabalho foi feita da seguinte forma: após o usuário efetuar o pedido e a entrega ser concluída no sistema, o usuário fornece uma nota de 1 a 5, para os itens/produtos comprados. Sendo a nota 1 ruim, 2 regular, 3 bom, 4 muito bom e nota 5 excelente. Assim é possível gerar uma lista de melhores e piores itens avaliados.

Figura 33: Avaliação dos Produtos pelos Usuários.

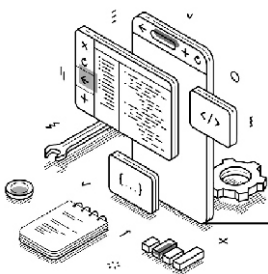
#	Produto	Nota
1	Marguerita	1
2	Calabresa	1

Status do Pedido: Efetuado

Voltar Gravar Notas

Fonte: Garbin & Silveira (2017)

Após o desenvolvimento do sistema, foi realizada a validação do mesmo junto ao gerente da Pizzaria Fratello, empresa localizada na cidade de Rodeio Bonito/RS. O gerente fez o acompanhamento das funcionalidades e do funcionamento do sistema. Após o acompanhamento da implementação do sistema, o gerente fez alguns elogios e sugestões. O mesmo acredita que este



tipo de sistema é um diferencial para sua empresa e também para a comodidade dos clientes, mantendo assim um laço entre clientes e a empresa.

Em um primeiro momento foi feita a demonstração do sistema localmente. Após a aprovação do mesmo foi feita a requisição para registro de domínio em nome da empresa na internet como também da hospedagem *web*. Para o aplicativo *mobile* foi realizada a requisição de cadastro no *Google Play* para a publicação do mesmo.

Pretendendo aplicar o protótipo de SR desenvolvido efetivamente para fins comerciais, alguns trabalhos futuros deverão ser elaborados, tais como: elaboração e adaptação da interface para pessoas com deficiência; elaboração de um módulo para reservas e eventos na sede da empresa; elaboração de um módulo para controle de estoque de produtos; elaboração de um módulo para controle financeiro sobre as vendas; elaboração de um melhor estudo e adaptação para sistemas de recomendação que utilizam técnicas matemáticas, tais como a correlação de *Pearson*, na filtragem colaborativa (Torres, 2004), integração com outras páginas e também sobre utilização de sistemas de recomendação pagos e *open-source*; e integração com sistemas de pagamentos *online*.

3.4 Framework Genérico de Recomendação para Lojas Virtuais (Prado & Silveira, 2014)

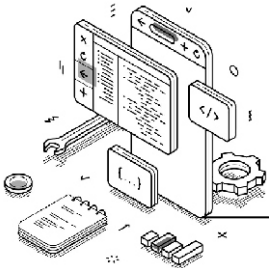
Diferentemente dos SRs apresentados nas seções anteriores deste capítulo, o trabalho de Prado & Silveira (2014) compreende um *framework* genérico, que pode ser integrado a um sistema de loja virtual já existente, por meio de configurações e parametrizações que serão apresentadas nesta seção.

A recomendação dos produtos foi realizada, inicialmente, por meio da aplicação de métodos de filtragem baseada em conteúdo. Podem ser citados alguns métodos de recomendação utilizados, tais como: recomendar produtos da mesma categoria de produtos já comprados, recomendar produtos na média de valores de produtos já comprados pelo cliente, recomendar produtos com maior quantidade em estoque, recomendar produtos mais recentes (que foram cadastrados recentemente no estoque), entre outras recomendações relevantes ao usuário.

Dentro deste contexto de recomendação, o administrador ou gerente da loja virtual pode selecionar parâmetros para gerar as recomendações de acordo com as suas necessidades, porém, estes parâmetros poderão ou não estar disponíveis de acordo com os campos disponibilizados pelo banco de dados da loja virtual.

O protótipo de *framework* foi desenvolvido utilizando-se a linguagem de programação PHP. O banco de dados utilizado no desenvolvimento foi o *MySQL*.

O protótipo de *framework* implementado foi validado primeiramente para lojas virtuais que utilizam banco de dados *MySQL*, por ser consideravelmente



simples a sua conexão com o PHP, e por ser um SGBD relacional bastante utilizado. Porém, a implementação e validação para outros bancos de dados pode ser apresentada como trabalhos futuros, com o intuito de deixar o *framework* o mais genérico possível.

Para integrar o *framework* de recomendação a uma loja virtual já existente é preciso realizar um processo de configuração do *framework*, o qual deve ser realizado apenas quando o usuário entra no sistema pela primeira vez. O processo de configuração do *framework* implementado é bastante simples, e consiste em 6 passos que são necessários para a conexão com a base de dados da loja virtual.

O primeiro passo do processo de configuração é digitar o nome da base de dados da loja virtual no campo solicitado e informar o sistema gerenciador de banco de dados, que, a princípio, foi implementado apenas para *MySQL*. O segundo passo consiste em informar os nomes de algumas tabelas encontradas no banco de dados, tais como, o nome da tabela de produtos e da tabela de clientes. Os outros 4 passos restantes consistem em informar campos de acordo com as tabelas informadas no segundo passo, sendo estes campos carregados automaticamente do banco de dados, restando ao usuário apenas a tarefa de selecionar o nome dos campos correspondentes.

A Figura 34 apresenta o segundo passo do processo de configuração do *framework* que, como citado anteriormente, consiste em informar os nomes das tabelas encontradas na base de dados da loja virtual.

Figura 34: Configuração do *Framework*.

A interface de configuração do Framework, etapa (2/6) Dados das Tabelas, apresenta um formulário com os seguintes campos:

- * Nome da Tabela de Produtos:
- Nome da Tabela de Clientes:
- Nome da Tabela de Vendas:
- Nome da Tabela de Vendas/Produtos:

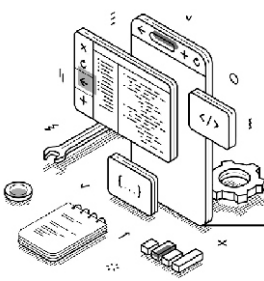
Obs.: Campos com (*) são de preenchimento obrigatório.

Botão:

Fonte: Prado & Silveira (2014)

Como podemos observar na Figura 34 o processo de configuração do *framework* é bastante simples e consiste apenas em preencher alguns campos necessários para a conexão com a base de dados da loja virtual. Podemos observar, também, que alguns campos são de preenchimento obrigatório e estão marcados com um asterisco, sendo os outros de preenchimento opcional, porém, o não preenchimento dos mesmos pode ocasionar no 'bloqueio' de alguns métodos de recomendação.

A parametrização das recomendações é disponibilizada no *framework* por meio de uma interface gráfica simples e clara, onde o administrador da loja



virtual pode acessar com os devidos dados de usuário do *framework* (*login*, senha e permissão de acesso). Na tela de parametrizações são apresentados os possíveis parâmetros que o usuário do *framework* pode selecionar para a geração das recomendações para os clientes da loja virtual. Após selecionar os parâmetros desejados, o usuário deverá clicar no botão salvar para gravar as informações no banco de dados do *framework*, que serão usados posteriormente para gerar as devidas recomendações.

Para melhor entender o seu funcionamento, a Figura 35 apresenta a tela de seleção de parâmetros de recomendações, a qual apresenta os títulos das recomendações bem como a descrição das suas funcionalidades.

Figura 35: Parametrização das Recomendações.

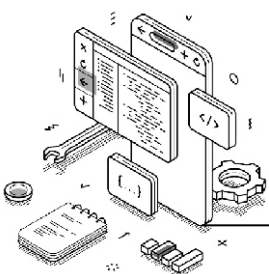


Fonte: Prado & Silveira (2014)

Pode ocorrer o caso de haver algum parâmetro que esteja bloqueado para o usuário selecionar. Isto pode ocorrer quando no momento de preencher os dados de configuração o usuário se esqueça de preencher algum campo, ou deixar de preencher algum campo por motivo do mesmo não existir no banco de dados da loja virtual, ou mesmo por motivos próprios, impossibilitando a aplicação de alguns métodos de recomendação pela falta de acesso aos dados necessários.

O usuário do *framework* pode selecionar parâmetros tais como: recomendar produtos que estão há mais tempo no estoque, recomendar produtos que estejam na mesma categoria de produtos que o usuário já comprou, recomendar produtos com maior quantidade em estoque, recomendar produtos na média de preço de outros produtos já comprados pelo cliente, entre outros.

A integração da loja virtual com o *framework* desenvolvido é realizada por meio da linguagem de programação PHP, sendo que o desenvolvedor da loja virtual pode fazer uma chamada direta ao método de recomendação do *framework*, sendo necessário incluir o arquivo 'recomende.php' e chamar a função *recomende*, passando como parâmetros o código do cliente da loja virtual e a quantidade de produtos que deseja recomendar.

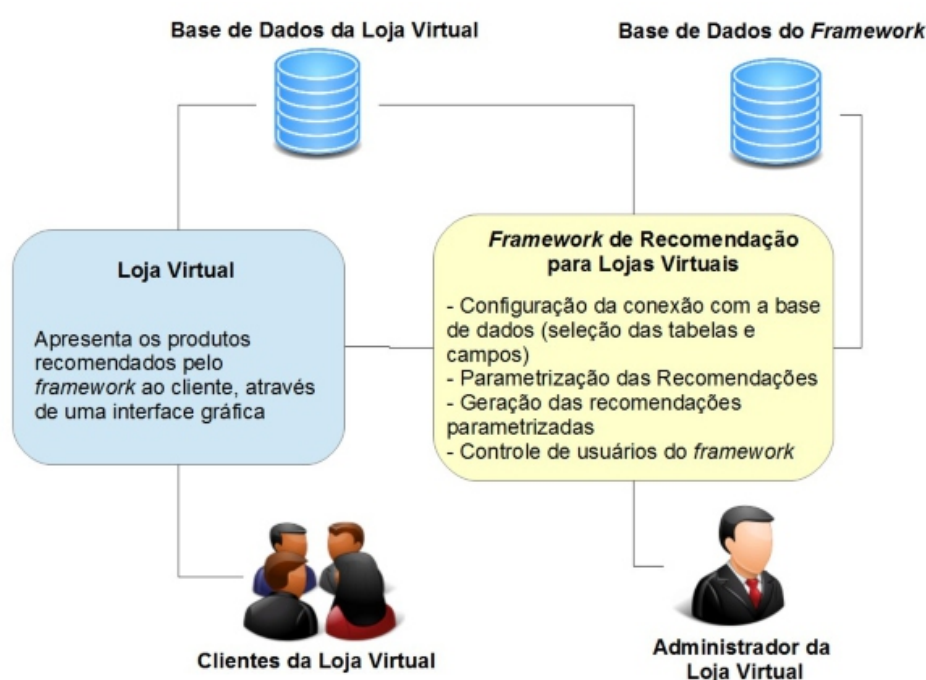


Caso a loja virtual seja desenvolvida em outra linguagem de programação, disponibilizamos outro meio de receber os produtos recomendados, por meio do *Javascript Object Notation* (JSON), que é uma formatação de troca de dados alternativa para o *Extensible Markup Language* (XML). Por meio dela o usuário poderá fazer uma chamada ao endereço URL (*Uniform Resource Locator*) do arquivo 'recomende.php', passando por meio do método GET o código do cliente e a quantidade de recomendações, o qual irá retornar um JSON formatado com os códigos dos produtos que devem ser recomendados.

Após realizar a chamada da função, o *framework* irá checar qual método de recomendação está selecionado e irá gerar a recomendação de acordo com o mesmo, retornando um vetor com os códigos dos produtos recomendados. Com este vetor com os códigos dos produtos, o desenvolvedor da loja virtual irá carregar os produtos da sua base de dados e apresentar na loja virtual do modo que melhor lhe convier.

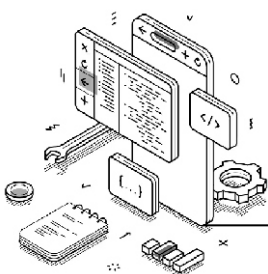
O esquema da Figura 36 apresenta, em alto nível, o funcionamento e a interação do *framework* com a loja virtual, bem como a interação com as bases de dados e com os agentes envolvidos, buscando facilitar o entendimento do mesmo.

Figura 36: Funcionamento do *Framework*.



Fonte: Prado & Silveira (2014)

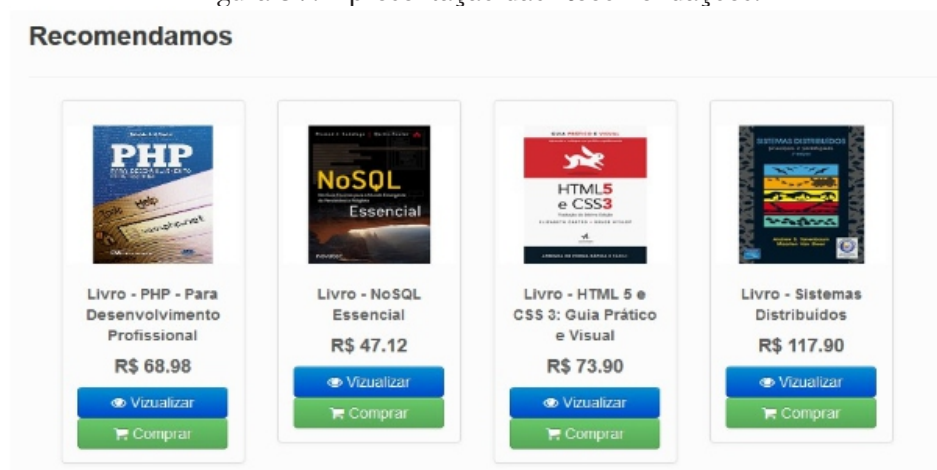
O *framework* implementado foi testado a partir de uma loja virtual desenvolvida especialmente para este fim, a qual possui as principais tabelas e campos existentes na base de dados de uma loja virtual real. Por meio dos testes de integração entre o *framework* e a loja virtual desenvolvida, foi



possível notar a grande facilidade de implementação e configuração do *framework* com a loja virtual.

A Figura 37 apresenta a forma como os produtos recomendados pelo *framework* são apresentados ao cliente por meio da interface da loja virtual desenvolvida.

Figura 37: Apresentação das Recomendações.



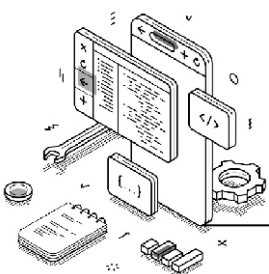
Fonte: Prado & Silveira (2014)

A validação do *framework* implementado se deu por meio de um questionário, o qual foi preenchido por usuários que possuem contato direto com lojas virtuais reais, possuindo assim um maior conhecimento de negócio e de vendas *online*. Além do propósito de permitir que fossem realizados testes, a loja virtual desenvolvida foi utilizada também no processo de validação, a qual serviu como meio de apresentação dos produtos recomendados pelo *framework* aos avaliadores, por meio de uma interface bastante semelhante a uma loja virtual real.

Observando os resultados do questionário de validação, foi possível analisar que todos os avaliadores concordam que sistemas de recomendação são muito importantes para lojas virtuais e podem aumentar significativamente o número de vendas. Todos os avaliadores entrevistados também concordaram que o *framework* implementado simplificará bastante o meio de gerar recomendações relevantes aos usuários, e afirmaram que utilizariam o *framework* em suas lojas virtuais.

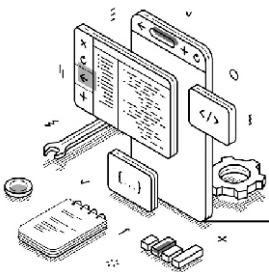
Com a aplicação do questionário de validação também foram feitas algumas sugestões de melhorias no *framework*, tais como:

- *Alteração na recomendação de produtos de valor semelhante*: a qual já foi implementada. Desta forma passou-se a recomendar produtos de valor médio entre os produtos já comprados pelo usuário, baseando-se em uma média aritmética simples e aplicando uma taxa de 20 por cento a mais e a menos do valor médio obtido, podendo esta porcentagem ser parametrizada de acordo com a necessidade de cada loja virtual;



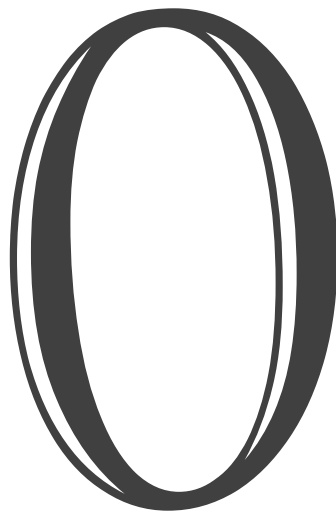
- *Acoplar a interface do framework a uma interface administrativa já utilizada pela loja virtual*, tornando assim o seu uso mais amigável ao usuário que já está familiarizado com a interface gráfica da administração da loja virtual. Como esta sugestão requer um estudo mais aprofundado sobre as plataformas já existentes no mercado, será incluída na lista de trabalhos futuros.

Durante o desenvolvimento do trabalho foram encontradas algumas dificuldades, tais como: a implementação da lógica das recomendações (os algoritmos de recomendação propriamente ditos) e como estabelecer a conexão com a base de dados da loja virtual de acordo com os parâmetros informados. O protótipo de *framework* foi validado primeiramente para lojas virtuais que possuem banco de dados *MySQL*, podendo-se assim citar como trabalhos futuros a implementação e validação para outros sistemas gerenciadores de bancos de dados, buscando tornar o *framework* o mais genérico possível.





EXPERIÊNCIAS DESENVOLVIDAS COM SISTEMAS HIPERMÍDIA ADAPTATIVOS

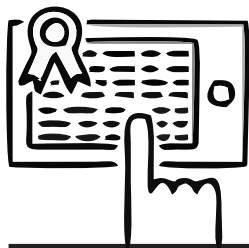
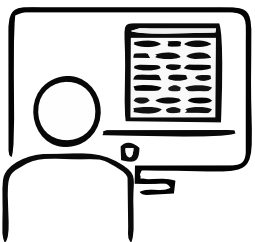
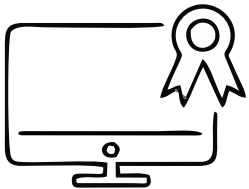


presente capítulo apresenta, inicialmente, conceitos compreendendo o desenvolvimento de SHA (Sistemas Hipermídia Adaptativos). Os Sistemas Hipermídia (que combinam Multimídia e Hipertexto) são capazes de armazenar um documento e permitir a leitura do mesmo de forma não convencional ou não linear, o que permite quebrar a rigidez do controle de outros sistemas que armazenam informações e/ou documentos.

Na área da Educação os Sistemas Hipermídia, muitas vezes, são incorporados a outros sistemas, tais como os STIs (Sistemas Tutores Inteligentes), para que os conteúdos a serem estudados possam ser visualizados de forma não-linear (Coelho, 1995).

Existe uma distinção sobre os tipos de Sistemas Hipermídia que são capazes de realizar alguma forma de personalização, podendo ser customizáveis e também chamados de adaptáveis ou adaptativos, onde um sistema pode ser personalizado e outro adaptativo. Para realizar a personalização, o usuário possui importância fundamental para o sistema, pois somente através dele é possível realizar a coleta dos dados sobre o mesmo, com o uso desses dados é que serão elaborados os modelos de usuário. A diferença entre os modelos é a maneira como a adaptação do conteúdo é feita (Koch, 2001).

Koch (2001) explica que “SHA podem ser sistemas de hipermídia como sistemas adaptativos. Através da combinação da hipermídia com os Sistemas Tutores Inteligentes (STI) com base na adaptação da informação, da interface e layout exibidos, ou pela forma de visualização que é dada a informação, ou seja, como a navegação é proposta”. Brusilovsky (1995) diz que todo Sistema Hipermídia de certa forma é um sistema adaptativo, dando possibilidade aos usuários de determinar o caminho que desejam seguir dentro da aplicação. No âmbito



educacional, por exemplo, para facilitar esse processo, um Sistema Hipermídia precisa ter determinadas informações sobre o conhecimento que o aluno já possui sobre o conteúdo que irá estudar, determinando qual caminho mais eficiente para que o conteúdo seja apresentado de maneira adaptativa ao que está sendo ensinado. Tem-se então um novo modelo de Sistema Hipermídia, que é chamado de Sistema Hipermídia Adaptativo (SHA).

Os SHA podem ser aplicados em diferentes áreas, tendo seu maior foco de atuação na área de Educação, onde as técnicas de hipermídia apresentam maior efetividade. O uso de técnicas de Hipermídia Adaptativa maximiza a possibilidade de aprendizagem, pois acabam acelerando e concentrando as informações com maior eficácia quando se trata de atividades desenvolvidas na modalidade de EaD (Lorenzi & Silveira 2011; Falkembach & Tarouco, 2000). Conforme Brusilovsky (1995), um SHA deve seguir três princípios fundamentais: 1) ser um sistema de hipermídia ou de hipertextos; 2) possuir um sistema de modelo de usuário e 3) deve possuir a capacidade de se adaptar ao usuário por meio desse modelo.

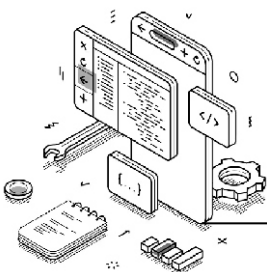
Os SHA focam em personalizar o sistema hipermídia por meio das características individuais de cada usuário. Os SHA realizam a coleta de dados com base no perfil do usuário, com intuito de estruturar um modelo de usuário baseado em suas características, suprimindo suas necessidades com a adaptação do conteúdo e/ou da interface.

Um dos objetivos dos SHA é a personalização da interface para cada usuário, de acordo com suas necessidades, conhecimento, modelo cognitivo e características específicas. Dessa forma os SHA tendem a moldar a interface para cada usuário utilizando a melhor opção entre as possibilidades existentes, apresentando apenas o melhor conteúdo já adaptado para aquele determinado usuário (Palazzo, 2000; Palazzo, 2002).

Entre as vantagens de um SHA, estão a de facilitar e agilizar o material a ser ofertado ao usuário. Dessa forma é necessário antecipar suas expectativas com a utilização do modelo que está representando seu perfil. Para alcançar o objetivo principal de um SHA é necessário que a apresentação do conteúdo seja em um nível semelhante ao grau de conhecimento do usuário e de forma interessante, didática e eficiente, assim elevando e maximizando o processo de aprendizagem, sem nunca deixar de lado o modelo de usuário o qual o sistema utiliza como referência para a adaptação do ambiente (Falkembach & Tarouco, 2000; Gasparini, 2003).

Duas categorias de adaptação que podem ser aplicadas em SHAs são propostas por Brusilovsky (1995):

- Apresentação Adaptativa: trata o conteúdo que irá ser apresentado para o usuário, de acordo com seus objetivos e seu conhecimento anteriormente adquiridos no assunto coberto pelo sistema. Com essas informações o sistema passa a exibir somente conteúdos relevantes e dentro dos níveis de conhecimento de cada usuário. Um exemplo de apresentação adaptativa é o sistema PUSH (Espizona & Höök, 1996).



- **Navegação Adaptativa:** tem o objetivo de auxiliar o usuário durante o caminho de busca por informações. Durante pesquisas envolvidas com páginas, somente assuntos realmente relacionados com o tema são ofertados, garantindo maior eficácia em sua procura. Uma das técnicas utilizadas é a de Anotação Visual de Elos, que tem como objetivo destacar um determinado elo, que no caso estaria sendo representado por uma âncora, por meio de ícones, cores, etc., conforme o atual estado mental do estudante.

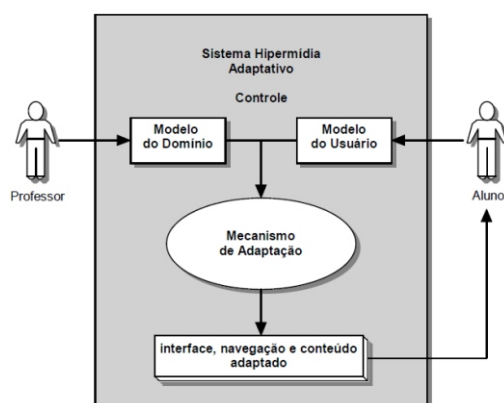
O modelo do usuário, em um SHA, é representado por toda a informação que o sistema possui sobre o usuário. No caso de um sistema na área educacional (como é a proposta deste trabalho), este usuário é o aluno. Sendo assim tem-se um modelo de aluno (conforme exemplo utilizado no SR apresentado no capítulo anterior, implementado por Colli & Silveira, 2023).

A manutenção dos modelos é feita pelo próprio sistema, porém o usuário tem a possibilidade de atualizar seu perfil e redefinir seus dados. Outras informações são adicionadas ao perfil do usuário conforme suas atividades e pesquisas realizadas, sendo que quando as mesmas são completadas, são adicionadas ao perfil do usuário (Falkembach & Tarouco, 2000).

Os modelos de usuário utilizados em SHA determinam um conjunto de dados para elaborar e gerar a interface para o usuário em tempo de execução (Gasparini, 2003).

O mecanismo de adaptação utiliza o modelo do usuário e o modelo do domínio para prover a adaptação do sistema dinamicamente. Modelos de domínio realizam a descrição de como a informação será estruturada conceitualmente, descrevendo o domínio da aplicação, utilizando páginas e conceitos que foram previamente definidos pelo autor (Wu, 2001). O modelo de domínio pode receber o nome de base de conhecimento do domínio, e é nesse local que está representado o material que será utilizado como método instrucional, mais especificamente o conteúdo que será ministrado pelo SHA (Vicari, 2000).

Figura 38: Mecanismo de Adaptação de um SHA.



Fonte: Adaptado de Gasparini (2003)

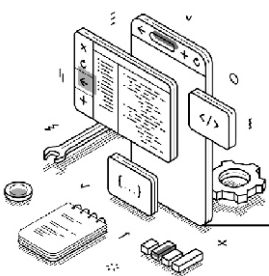
A Figura 38 apresenta a arquitetura básica de um SHA, onde o professor elabora e insere os conteúdos das disciplinas e demais informações necessárias, o aluno define seu modelo de usuário por meio de questionários, nível de conhecimento, estilos cognitivos, entre outros. Após a definição do modelo de usuário, os mecanismos de adaptação de conteúdo são aplicados, com isso as alterações na interface, navegação e conteúdo são feitas para que o aluno receba o conteúdo de forma adaptativo ao seu modelo de usuário.

Existe uma grande diversidade de modelos para a aplicação de táticas de ensino. O modelo de hipertexto é aquele em que o estudante navega em uma estrutura de hipertexto de maneira livre, explorando e visualizando o conteúdo de forma aleatória, utilizando apenas como base seus desejos e objetivos. Cada subdivisão lógica dos assuntos pesquisados estão ligados por diversos e diferentes *links*, o que amplia as possibilidades de navegação e exploração do domínio. O modelo de hipertexto vem a ser um excelente meio para aplicação e construção de STI e também de SHA, já que pode oferecer diferentes formas de representação de conhecimento no mesmo documento, dando ao aluno mais espaço na participação em sua trajetória de aquisição do conhecimento enquanto é orientado no aspecto pedagógico de forma diversificada (Gasparini, 2003).

4.1 Adaptação de Cursos a Distância empregando Objetos de Aprendizagem (Silveira et al., 2016)

Esta seção apresenta o trabalho desenvolvido por Silveira *et al.* (2016), compreendendo uma proposta de arquitetura de adaptação de cursos na modalidade de EaD. Acreditamos que os alunos de cursos disponibilizados na modalidade de EaD, conforme a área do conhecimento, possuam estilos cognitivos/estilos de aprendizagem diferenciados (Geller 2004; Silveira, 2006). Neste sentido, esta seção apresenta a definição de uma arquitetura que permita a adaptação de materiais didáticos digitais (Objetos de Aprendizagem) que possam ser disponibilizados em cursos ministrados na modalidade de EaD, por meio da aplicação de técnicas de Hipermídia Adaptativa, aliadas à definição de um modelo de aluno.

Esta arquitetura permitirá que docentes possam desenvolver e reutilizar materiais didáticos digitais e que os discentes tenham acesso a conteúdos diferenciados, adaptados de acordo com seus estilos cognitivos. Definimos uma arquitetura baseada em ferramentas livres, que possa ser aplicada por diversas instituições de ensino, possibilitando que sejam realizadas adaptações, tanto em nível de interface bem como de conteúdo, conforme o perfil/curso dos usuários deste ambiente. As adaptações serão realizadas por meio do emprego de técnicas de apresentação e de navegação adaptativa. Neste contexto, acreditamos que, com base nos estilos cognitivos dos alunos, seja possível reutilizar OAs existentes, disponibilizando-os em um repositório e armazenando-os de forma padronizada, permitindo que sejam empregadas diferentes técnicas de hipermídia adaptativa para visualização e navegação em cursos ministrados na modalidade de EaD (Souza *et al.*, 2015).



A Figura 39 apresenta a arquitetura proposta, dividida em três camadas: repositório de conteúdos (onde serão armazenados os objetos de aprendizagem e outros materiais didáticos digitais), SHA (onde serão armazenadas as informações dos estilos cognitivos dos alunos e as técnicas de adaptação) e o ambiente virtual de aprendizagem, onde os conteúdos e resultados da aplicação das técnicas adaptativas serão visualizados.

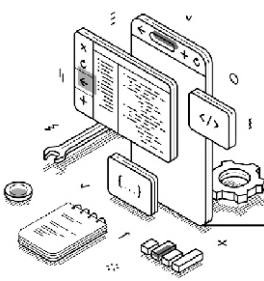
Os protótipos implementados e/ou em desenvolvimento, ligados a esta arquitetura, estão utilizando o Ambiente Virtual de Aprendizagem *Moodle*, já que o mesmo é largamente utilizado em diferentes Instituições de Ensino Superior, incluindo-se a UFSM. A implementação das técnicas de adaptação e do acesso ao repositório de conteúdos será realizada por meio da linguagem de programação PHP e o Sistema Gerenciador de Bancos de Dados *MySQL*, visando facilitar a integração com o AVA *Moodle*, que é implementado nesta mesma plataforma.

O modelo de aluno será baseado na aplicação do instrumento elaborado e validado por Bariani (1998) e implementado computacionalmente por Geller (2004), que compreende os alunos em seis estilos cognitivos: convergente, divergente, holista, serialista, reflexivo e impulsivo. A partir da aplicação de um questionário inicial, proposto por Bariani (1998), a camada da arquitetura correspondente ao SHA poderá definir quais OAs devem ser apresentados para o aluno, por meio de técnicas de apresentação adaptativa e de ocultação. O Quadro 4 apresenta as formas de apresentação de conteúdos e instrumentos de avaliação que foram definidos para utilização na arquitetura proposta, com base no trabalho realizado por Silveira (2006). Para identificar os OAs de acordo com as formas detalhadas no Quadro 4, é preciso utilizar os metadados dos mesmos, conforme regras para armazenamento dos OAs no repositório de conteúdos.

Quadro 4: Estilos Cognitivos, Tipos de Conteúdo e Instrumentos de Avaliação

Estilo Cognitivo	Conteúdo	Avaliação
Convergente	Artigos, tutoriais, apostilas, fotos, imagens animadas, desenhos	Desafios
Divergente	Tópicos, <i>links</i> , sites de busca, gráficos, diagramas	Fórum, desafios
Holista	Artigos, <i>links</i> , sites de busca, livros <i>on-line</i> , diagramas	Bate-papo (<i>chat</i>), pesquisa
Impulsivo	Tópicos	Bate-papo (<i>chat</i>)
Reflexivo	Tutoriais, artigos, livros, capítulos de livros, gráficos, diagramas	Fórum, questionário, resenha
Serialista	Apostilas, tutoriais, tópicos, gráficos, desenhos sequenciais	Fórum, questionário

Fonte: Geller (2004); Silveira (2006)

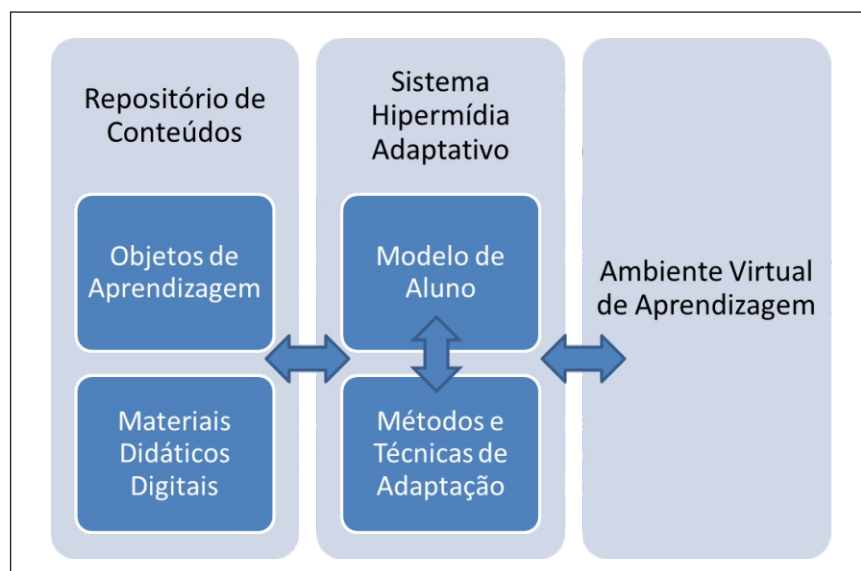


Como a proposta da arquitetura é baseada em *software* livre, ao estudar ferramentas para a construção de OAs (que serão armazenados no repositório de conteúdos), optou-se por desenvolvê-los utilizando HTML 5 e *JavaScript* (Meyer, 2011) e a ferramenta *Ardora* (Matanza, 2014). De acordo com o nível de conhecimento em informática, os docentes poderão optar por desenvolver seus OAs utilizando a ferramenta *Ardora*, que não necessita de programação (código-fonte) ou por meio de programação com HTML5 e *JavaScript*.

Considerando os objetivos propostos, verifica-se que existem dois momentos distintos na realização deste projeto de pesquisa. Inicialmente será empregada a metodologia *dissertação-projeto* (Ribeiro & Zabadal, 2010). Nesta primeira fase serão estudados os aspectos teóricos que irão fundamentar o projeto, bem como serão definidas tecnologias e técnicas a serem empregadas. A primeira fase resultará na definição e implementação da arquitetura adaptativa proposta, constituindo-se em um protótipo que será validado por meio de um estudo de caso.

Em um segundo momento será realizado um estudo de caso (Yin, 2015), por meio do qual a arquitetura adaptativa proposta será aplicada para a disponibilização de cursos na modalidade a distância, validando-a com a participação de professores e alunos. A Figura 39 apresenta a arquitetura proposta.

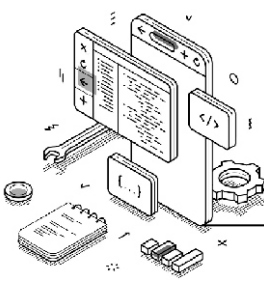
Figura 39: Arquitetura Proposta.



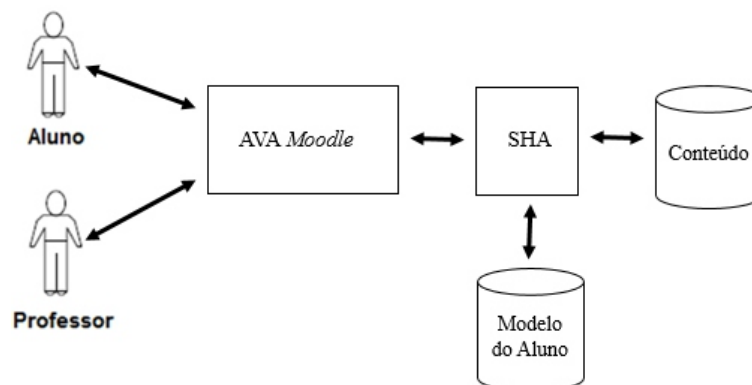
Fonte: Silveira *et al.*, 2016.

De acordo com a arquitetura definida na Figura 38, está sendo desenvolvido um protótipo para criar o SHA, que será integrado ao AVA *Moodle*.

A Figura 40 apresenta a arquitetura de alto nível do SHA, envolvendo o Professor, Aluno, AVA *Moodle*, Sistema Hipermídia Adaptativo (SHA), conteúdo a ser adaptado e o Modelo de usuário (Aluno).



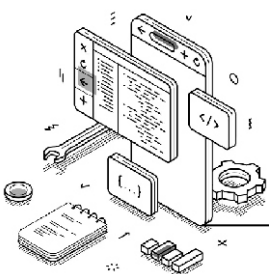
Arquitectura de alto nivel



Com base no levantamento e estudo sobre as diversas técnicas existentes na navegação adaptativa, serão aplicadas para a elaboração do SHA proposto as técnicas de orientação direta, ocultação de *links*, classificação adaptativa e anotação adaptativa (Brusilovsky, 1996; De Bra, 2000, Gasparini, 2003, Palazzo, 2002; Souza *et al.*, 2015):

- Orientação direta: apresentar ao aluno as tarefas pendentes ao acessar o sistema (com determinado tempo de antecedência);
- Orientação direta: apresentar ao aluno a tarefa que deve ser entregue naquela data;
- Anotação Adaptativa: destacar ou marcar os *links* referentes aos tipos de materiais mais indicados de acordo com o estilo cognitivo do aluno;
- Anotação Adaptativa: destacar os *links* (materiais) que ainda não foram vistos pelos alunos;
- Anotação Adaptativa: inserir uma marcação indicando o tipo do material postado pelo professor;
- Anotação Adaptativa: destacar novos materiais inseridos pelos professores;
- Classificação Adaptativa: colocar os materiais em ordem (os mais indicados para o estilo cognitivo devem aparecer primeiro);
- Ocultação Adaptativa: esconder materiais que não devem ser vistos, de acordo com o estilo cognitivo do aluno.

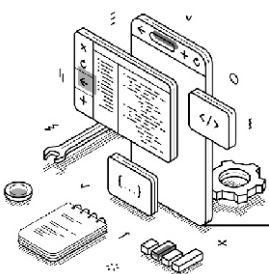
As principais contribuições científicas e tecnológicas do referido projeto consistem na definição de uma arquitetura adaptativa, que permita a reutilização de objetos de aprendizagem e outros materiais didáticos digitais e a implementação de técnicas de hipermídia adaptativa em um AVA, que



poderá ser aplicada em diferentes Instituições de Ensino Superior, já que se pretende utilizar a filosofia de *software* livre, o que possibilita que os resultados oriundos do projeto sejam difundidos através da comunidade acadêmica.

Acreditamos que o uso de técnicas de Hipermissão Adaptativa ampliará as possibilidades de aprendizagem e potencializará o processo de ensino, em especial, a interatividade necessária nas atividades realizadas a distância. Esta arquitetura poderá ser empregada, também, em atividades semipresenciais, em cursos que adotam essa estratégia pedagógica.

Atualmente o grupo de pesquisa está implementando um sistema de recomendação de objetos de aprendizagem, utilizando a lógica *fuzzy*, que será incorporado à arquitetura proposta, integrado à camada do SHA (que também encontra-se em desenvolvimento).





EXPERIÊNCIAS DESENVOLVIDAS COM RACIOCÍNIO BASEADO EM CASOS

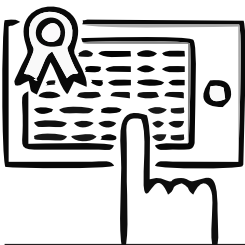
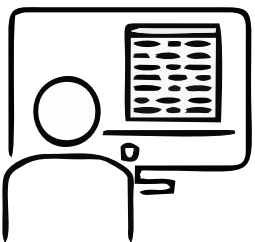
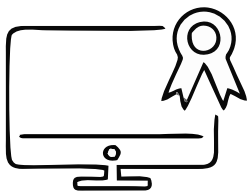
D

e acordo com Lagemann (Lagemann, 1998
apud Fernandes, 2005):

“A origem do Raciocínio Baseado em Casos data de 1977, devido a uma pesquisa na área da ciência cognitiva, desenvolvida por Schank e Abelson. O desenvolvimento do RBC foi estimulado pelo desejo de compreender como as pessoas conseguem recuperar informações e que elas, frequentemente, resolvem problemas lembrando de como solucionaram casos similares no passado” (p. 28).

Em síntese, o RBC é um instrumento de raciocínio da Inteligência Artificial (IA), que tem como finalidade buscar a solução de um problema através da adaptação de uma experiência passada parecida (Barone, 2003). Wangenheim e Wangenheim (2003) definem o RBC como uma solução que resolve futuros problemas por meio da recuperação e adaptação de soluções passadas. O RBC tem como finalidade resolver problemas ao recuperar e adaptar experiências passadas – chamadas casos – arquivados em uma base de casos (Ferrari *et al.*, 2019). Segundo Fernandes (2005, p. 27), “o processo característico do RBC consiste em: identificar o problema atual, buscar a experiência mais semelhante na memória e aplicar o conhecimento dessa experiência passada no problema atual”.

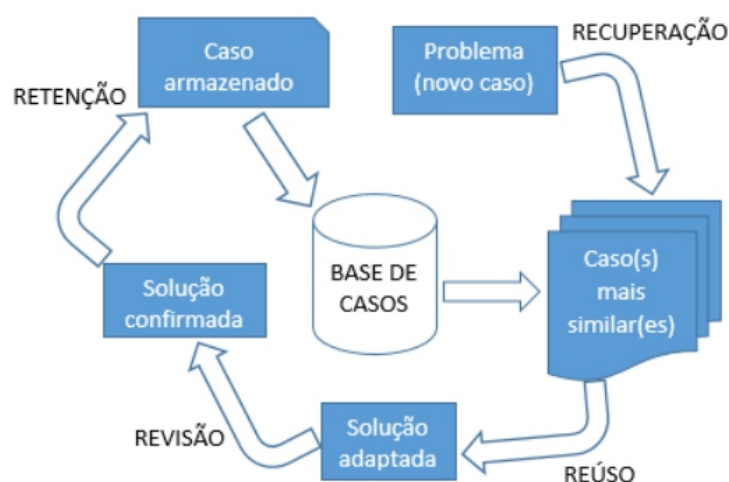
Wangenheim e Wangenheim (2003) descrevem que a técnica de RBC tem sido muito utilizada no desenvolvimento de aplicações de sistemas baseados em casos reais, visto que, esta técnica é muito ampla, pois ela pode ser aplicada em diversos campos, sendo eles: diagnóstico médico, sistema de suporte de software, planejamento, gestão do conhecimento, entre outros.



O modelo mais aceito para o processo RBC é o Ciclo de RBC proposto por Aamondt e Plaza (Aamondt & Plaza, 1994 *apud* Wangenheim & Wangenheim, 2003), que engloba um ciclo de raciocínio contínuo composto por quatro tarefas principais (como mostra a Figura 41):

- Recuperar: Nesta etapa recupera-se diretamente na base de casos, o caso mais semelhante ao novo problema. Por meio da identificação e da pesquisa de índices, efetua-se o cálculo de similaridade entre o caso recuperado e o novo problema (Barone, 2003);
- Reutilizar: esta etapa envolve a reutilização da solução relacionada ao caso recuperado. A solução do problema recuperado é geralmente utilizada como ponto de partida para propor uma solução para o novo problema. Essa etapa também é denominada adaptação de casos (Carvalho *et al.*, 2011);
- Revisar: quando não se aplica diretamente a solução ao novo problema, necessita-se revisar a solução do caso recuperado, gerada pelo processo de reutilização. Analisam-se as desigualdades entre o problema recuperado da base de casos e o problema de entrada, e qual elemento do caso recuperado pode ser inserido para o novo caso, adequando a solução deste caso à solução do novo problema (Barone, 2003). Segundo Lagemann (Lagemann, 1998 *apud* Fernandes, 2005), o processo de revisão pode ser executado de diversas formas: inclusão de um novo comportamento à solução recuperada; eliminação de um comportamento da solução recuperada ou a substituição de parte de um comportamento;
- Reter: Se após efetuar a revisão, for expressiva a solução adaptada, o problema e sua solução podem ser guardados na base de conhecimento ou base de casos (Carvalho *et al.*, 2011).

Figura 41: Ciclo do Raciocínio Baseado em Casos.



Fonte: Adaptado de Wangenheim & Wangenheim (2003).

Sempre que um desenvolvedor de *software* encontrar acessível uma base de casos para ser analisada, será possível examiná-la para extrair conhecimento a ser utilizado na tomada de novas decisões. Para efetuar a construção de um sistema RBC a partir de uma base de casos é necessário passar por descrições técnicas e linhas de implementação de cada um dos componentes do sistema (Ferrari *et al.*, 2019). Segundo Wangenheim e Wangenheim (2003), as etapas mais significativas do processo de desenvolvimento de um sistema RBC são a representação de casos, indexação, recuperação de casos e adaptação de casos, que serão descritas a seguir.

A etapa de representação de casos em um sistema de RBC é uma atividade de grande importância, pois é por meio dela, que se pode atingir bons resultados na aplicação desta técnica para a resolução de problemas (Aamondt & Plaza, 1994 apud Thé, 2011). Wangenheim e Wangenheim (2003) relatam que caso é um elemento de conhecimento contextualizado interpretando uma experiência real ou episódio concreto, composto por uma lição já vivenciada, que é o conteúdo e a circunstância em que a lição poder ser usada ou aplicada.

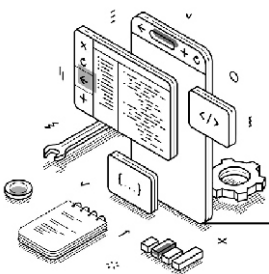
Um caso geralmente representa a descrição de uma situação que pode ser chamado de problema, juntamente com as experiências adquiridas que será a solução para resolver o problema e finaliza com uma conclusão. Sendo assim, um caso é composto por três componentes (Vitorino, 2009):

- Descrição do problema: descreve o problema que foi solucionado ou a condição que foi assimilada;
- Descrição da solução: descreve a solução que deve ser aplicada para um problema específico em sua descrição;
- Conclusão: é a avaliação da solução aplicada para resolver determinado problema.

Segundo Lee (Lee, 1998 *apud* Fernandes, 2005) a indexação é uma parte essencial do RBC, pois a mesma aconselha a avaliação de similaridade. A similaridade relaciona-se à análise comparativa entre o novo caso de entrada e os casos existentes na base para determinar o quanto eles são similares. Sendo assim, os índices são usados para definir o grau de similaridade dos casos. Os índices de um caso são combinações de seus atributos mais significativos, que possibilitam diferenciá-los de outros e mencionar quais casos são úteis para um determinado problema descrito (Wangenheim & Wangeheim, 2003).

A indexação dos casos provém da compreensão do conteúdo e da finalidade da informação que eles armazenam. Muitas diretrizes na indexação têm sido sugeridas por pesquisadores de RBC (Watson & Marir, 1994 *apud* Barone, 2003). Os índices devem permitir identificar, primeiramente, de que maneira a informação vai ser restaurada na solução de diferentes problemas; endereçar as similaridades úteis entre os casos; serem palpáveis para serem reconhecidos no futuro; além de permitir uma ampliação futura na base de casos (Barone, 2003).

O processo de recuperação de casos começa por meio da descrição de um problema e se dá como encerrado quando um melhor caso for localizado. O sistema verifica na base de casos, o caso mais parecido com o novo problema



(Riesbeck, 1999 *apud* Vitorino, 2009). A etapa de recuperação de casos é dividida em três subtarefas: identificação das características, comparação e seleção; que devem ser executadas nesta ordem (Aamodt & Plaza, 1994 *apud* Wangenheim & Wangenheim, 2003):

- Identificação das características: indicar ao sistema quais as características do caso atual;
- Comparação: retornar da base de casos um conjunto de casos parecidos ao problema de entrada;
- Seleção: selecionar o caso com maior similaridade entre os casos do conjunto, obtidos no processo de comparação.

Segundo Lorenzi e Silveira (2011) os métodos mais conhecidos para recuperação de casos são o vizinho mais próximo, o método de recuperação indutivo e a recuperação baseada em conhecimento:

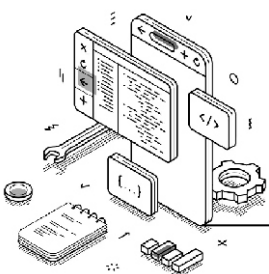
- Vizinho mais próximo: combina casos recuperados com base no somatório de pesos das características do novo problema. Os casos com o total de comparações com alguma similaridade métrica são retornados do processo de comparação;
- Método de Recuperação Indutivo: torna-se o melhor método quando a meta da recuperação é bem definida. Casos são indexados com base nas características mais importantes. A árvore resultante provê tempos de recuperação mais rápidos do que o método de recuperação do Vizinho Mais Próximo;
- Recuperação Baseada em Conhecimento: aplica o conhecimento do domínio para localizar casos relevantes. Esta abordagem é semelhante a sistemas especialistas baseados em regras, nos quais um especialista determina as características usadas para classificar os casos. O conhecimento não precisa ser completo.

Para iniciar a etapa de adaptação em um sistema de RBC faz-se a recuperação de um caso, a partir disso é realizada a reutilização de um caso arquivado na memória (Watson, 1997; Rafter, 2001 *apud* Thé, 2001).

Diante disso, Lee (Lee, 1996, *apud* Fernandes, 2005) destaca que a adaptação tem como finalidade alterar qualquer caso, quando for necessário, a fim de solucionar o novo problema, examinando-se as diferenças entre os casos recuperados e o problema inicial.

Vitorino (2009) destaca que existem várias técnicas de adaptação de casos, que combinam com a adaptação estrutural e derivacional, que são: adaptação nula; ajuste por parâmetro; abstração e reespecialização; adaptação baseada em crítica; reinstanciamento; substituição derivacional; reparo guiado por modelo; substituição baseada em casos.

Segundo Luger (2013) e Thé (2001) podem-se citar as seguintes vantagens do RBC:



- Habilidade de codificar diretamente conhecimento histórico: Por meio de histórias de casos disponíveis, relatos de reparos e outras fontes, novos casos podem ser criados, sem que seja necessário adquirir conhecimento intensivo de um especialista humano;
- Aceita atalhos no raciocínio: Ao encontrar um caso adequado, novos problemas podem ser solucionados em um menor tempo em relação a uma solução baseada em regras ou em modelos;
- Não é necessária uma análise extensiva do conhecimento do domínio. Já em um sistema baseado em regras é necessário que o engenheiro do conhecimento antecipe as interações das regras;
- Estratégias de indexação apropriadas aumentam a capacidade de compreensão e o poder de solução de problemas;
- Agilidade no desenvolvimento de aplicação e rapidez no processo de raciocínio;
- Maior facilidade para a manutenção e expansão de casos.

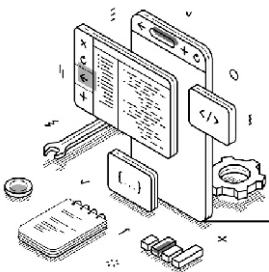
Como limitações, Luger (2013) destaca que os casos frequentemente não acrescentam um conhecimento profundo do domínio, o que pode ocasionar em casos mal aplicados, aconselhamentos errados ou de menor qualidade. Além disso, definir bons critérios para indexar e realizar casamento de casos é uma tarefa complexa. Laight (Laight, 1999 *apud* Thé, 2001) aborda algumas desvantagens relacionadas à utilização do RBC, sendo elas: elevado custo das ferramentas de teste para examinar sua eficácia; custo e complexidade na implementação do sistema RBC; existência de informações antigas na base de casos, que podem estar obsoletas; custo para efetuar a manutenção da base de casos, entre outras.

5.1 Ferramenta para Apoio ao Suporte Técnico em uma Rede FTTX empregando Raciocínio Baseado em Casos (Ferrari et al., 2019)

O trabalho desenvolvido por Ferrari *et al.* (2019) visou a atender uma necessidade de um provedor de acesso à Internet, de Frederico Westphalen/RS, permitindo uma centralização e melhor documentação da rede FTTx (*Fiber to the x*). Implementando um protótipo de ferramenta capaz de gerar relatórios a partir das informações armazenadas, além de auxiliar os técnicos nas soluções e/ou detecções de problemas dessa rede (visando à melhoria da capacitação dos mesmos), de forma mais precisa e ágil, com o emprego de técnicas de RBC.

A metodologia de pesquisa utilizada para o desenvolvimento desta ferramenta foi a Dissertação-Projeto, pois foi implementado um protótipo (Ribeiro & Zabadal, 2010).

A aquisição de conhecimento para a construção da base de casos da ferramenta implantada foi realizada por meio de conversas informais com os profissionais de diferentes departamentos (técnico de instalação, suporte



técnico, *Call Center* e administração de redes) que trabalham com a rede de FTTx no provedor de acesso à Internet. Cada profissional relatou as suas experiências vivenciadas no dia a dia para detectar e solucionar problemas encontrados nesse tipo de rede. Também foram verificadas informações em manuais de equipamentos ópticos utilizados no provedor, além do conhecimento das próprias experiências com redes FTTx.

Com base nestas conversas informais, os autores definiram que os casos (problemas que ocorrem nos equipamentos da rede FTTx) seriam armazenados em um banco de dados, contendo tabelas para armazenar o problema, suas diferentes características (com valores possíveis e pesos) e soluções propostas.

Para realizar a comparação dos casos de entrada (problemas) com os casos existentes na base, foi utilizado o método do *Vizinho mais próximo*. O algoritmo deste método constitui-se dos seguintes passos (Lorenzi & Silveira, 2011): 1) recebe uma base de casos L; 2) recebe um novo problema C; 3) calcula a similaridade entre C e os casos de L; 4) retorna o caso que é mais parecido com C. Além disso, deve-se aplicar a fórmula apresentada na Figura 42.

Figura 42: Fórmula – Vizinho mais Próximo.

$$\text{distância}(X,C) = (\sum w_f * \text{sim}(x_f, c_f)^2)^{1/2}$$

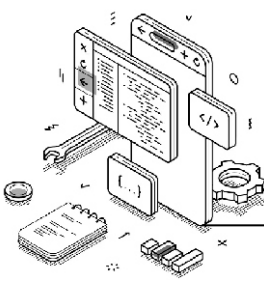
Fonte: Ferrari *et al.* (2019).

Este método foi escolhido tendo-se em vista a forma como os casos foram armazenados na base, sem a utilização de índices e/ou regras estabelecidas pelos especialistas, o que descarta a aplicação de outros métodos (Lorenzi & Silveira, 2011).

De acordo com a fórmula para o cálculo da distância, tem-se (Lorenzi & Silveira, 2011): C = novo caso ou novo problema; X = casos existentes na memória de casos; n = número de atributos; f = atributo individual variando de 1 a n; *sim* = função de similaridade para o atributo *f* nos casos X e C; *w* = peso do atributo *f*.

O algoritmo do vizinho mais próximo assume que cada caso $x = \{x_1, x_2, \dots, x_n\}$ é definido por um conjunto de *n* características *f*. Dado um novo problema C, a biblioteca de casos L e o peso *w* de cada característica *f*, o algoritmo recupera o caso mais parecido com o novo problema, ou seja, aquele que tiver a menor distância. A função de similaridade (*sim*) entre o novo problema e os casos da base será calculada da seguinte forma $|x_i - c_i|$, ou seja, a distância euclidiana entre os valores, considerando 0, quando os valores das características forem iguais ou 1, quando os valores comparados forem diferentes (Lorenzi & Silveira, 2011).

Por meio de conversas informais com profissionais de diferentes departamentos (técnico de instalação, suporte técnico, *Call Center* e



administração de redes) que trabalham com a rede de FTTx no provedor de acesso à Internet, cada profissional mencionou seus conhecimentos, suas principais necessidades e dificuldades, sendo elas técnicas ou teóricas sobre o assunto abordado. Neste sentido, constatou-se que cada departamento possui diferentes conhecimentos, necessidades e dificuldades. Juntamente com o departamento de administração de redes os autores verificaram os arquivos e documentos utilizados para realizar a documentação da rede FTTx, onde destacaram-se as principais informações que deviam ser inseridas na base de dados, para centralizar a documentação e permitir que fossem realizados relatórios. Por meio das informações coletadas, além do conhecimento das atividades realizadas diariamente, foram destacadas as principais necessidades ressaltadas durante o desenvolvimento e aplicação do protótipo.

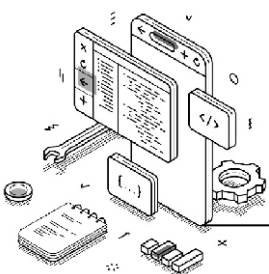
Desta forma, o sistema desenvolvido contém as seguintes funcionalidades: por meio de uma interface *web*, é capaz de efetuar consultas e gerar relatórios com os dados cadastrados dos equipamentos ópticos presentes na rede FTTx (entre a central e o ponto de atendimento óptico do cliente) e suas interligações. Por meio da utilização de técnicas de RBC (Raciocínio Baseado em Casos), esta ferramenta também possibilita auxílio aos profissionais dos Departamentos de Administração de Redes e *Call Center*.

Estes profissionais, ao interagir por meio do telefone, com os clientes e os demais departamentos externos (técnicos de instalação e suporte técnico) podem identificar e/ou corrigir problemas presentes na rede FTTx do provedor de acesso à Internet.

A ferramenta implementada utiliza a linguagem de programação PHP e o SGBD *MySQL*, visando facilitar o desenvolvimento. Para o gerenciamento da aplicação, utilizou-se o *framework CodeIgniter*, que permite o desenvolvimento de projetos de forma mais ágil, por meio do reaproveitamento de códigos. A ferramenta utiliza a abordagem MVC, *Model-View-Controller* (Modelo-Visão-Controle), que permite a separação entre a lógica e apresentação, bem como esta ferramenta é gratuita, leve e rápida. O MVC é um modelo de desenvolvimento e *design*, que tenta separar uma aplicação em três partes distintas. A *Model* está relacionada ao trabalho atual que a aplicação administra; a *View* está relacionada a exibir os dados ou informações da aplicação; e a *Controller* coordena partes anteriores, exibindo a interface correta ou executando algum trabalho que a aplicação precisa completar. Para o desenvolvimento da interface do protótipo foi utilizado o *framework front-end Bootstrap*, que permite um desenvolvimento *web* mais rápido e fácil.

A ferramenta permite o cadastro de OLT (*Optical Line Terminal*) e dos demais equipamentos existentes na rede FTTx. Além do cadastro de OLT, a ferramenta permite efetuar o cadastro de cidades que possuem a tecnologia FTTx, os *slots* pertencentes a cada OLT, os segmentos (HUB) da rede e os NAPs (*Network Access Point*). As funcionalidades implementadas permitem a realização de consultas, listagens das informações e a geração de relatórios.

Com relação à aplicação da técnica de RBC, existem dois momentos: 1) a manutenção da base de casos, por meio do cadastro de características, valores das características e problemas solucionados (casos) e 2) a entrada de novos



problemas e a busca por soluções. Primeiramente é realizado o cadastro dos valores que poderão ser associados às características. Por exemplo, a característica “*led(verde) power da ONU*” pode receber valores como: Ligado ou Desligado. Após realizar-se o cadastro dos valores permitidos para as características é possível realizar o cadastro das características, associando-as a seus possíveis valores. A Figura 43 mostra o processo de cadastro da característica “*led (verde) power ONU*” e a mesma é associada aos valores: Ligado e Desligado. Toda característica, quando cadastrada, deve ser associada a um ou mais valores. Também é necessário incluir uma descrição de fácil compreensão facilitando, assim, sua utilização, tanto ao ser associada a um problema, quanto a ser utilizada para descrever um novo problema, visando buscar uma solução.

Figura 43: Cadastro de Característica.

Nova Característica

Descrição da Característica:

Led (verde) Power da ONU

Valor Possível:

Inserir Valores

Lista de Valores Inseridos

Valores	
Desligado	Remover
Ligado	Remover

Enviar

Cancelar

Fonte: Ferrari *et al.* (2019).

Após ser realizado o cadastro de valores e de características, é possível inserir os casos na base, referentes a problemas já enfrentados pelos técnicos da empresa, bem como suas possíveis soluções. Ao efetuar a inserção de um novo caso é necessário descrever o problema, a solução do mesmo (podendo ser representada por um passo-a-passo para solucionar o problema, dependendo da situação do problema), inserir uma ou mais características, associando-as com o devido valor e peso (o somatório dos pesos não devem ultrapassar o valor 1, pois este é o limite utilizado no cálculo da similaridade) em relação ao problema a ser cadastrado. A Figura 44 mostra o processo de inserção de um problema já solucionado (um caso) na base de casos.

Após a base de casos possuir alguns problemas cadastrados é possível efetuar a etapa que envolve a busca de soluções para um novo problema relatado. Esta etapa consiste em descrever o novo problema por meio de características e valores cadastrados na base de casos.

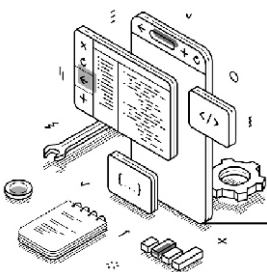


Figura 44: Cadastro de Problemas.

Cadastro Problema

Descrição do Problema:

ONU desligada

Características:

Valor Possível: Peso Característica:

Descrição da Solução:

Passo a passo a ser seguido:

- Verificar se a fonte da ONU está conectada na energia;
- Verificar se a fonte está plugada corretamente na ONU e se a mesma é compatível;
- Verificar se a ONU possui um botão que liga e desliga, certificar se o mesmo está ligado; Se mesmo assim a ONU não ligou;
- Testar a ONU com outra fonte, se não funcionar

Lista de Características Inseridas

Característica	Valor	Peso	Ação
Led (verde) Power da ONU	Desligado	1	Remover

Fonte: Ferrari *et al.* (2019).

A Figura 45 mostra um exemplo desta etapa, na qual o usuário selecionou as características e associou valores às mesmas. Ao efetuar a busca será executado o algoritmo com o cálculo do *vizinho mais próximo*, que verificará os casos mais semelhantes cadastrados na base de casos, retornando uma ou mais soluções possíveis, além de indicar o grau de similaridade em relação ao problema descrito e os problemas armazenados na base de casos.

Figura 45: Busca de Soluções.

Buscar Solução para o Problema

#	Característica	Valor
1	☐ Led (verde) Lan ONU	---
2	☐ Led (verde) PON da ONU	---
3	☐ Led (verde) Power da ONU	---
4	☐ Led (verde) Power Roteador	---
5	☐ Led (vermelho) LOS da ONU	---
6	☒ Nível de sinal óptico dentro do permitido	No Cliente
7	☒ Terminação óptica do cliente	Conectorizado

Fonte: Ferrari *et al.* (2019).

A Figura 46 mostra o resultado obtido a partir do exemplo da Figura 43. Neste exemplo foram retornadas 3 soluções, sendo que a primeira apresentada tem 100% de similaridade e, as 2 restantes, 50%. Quando existem várias soluções, as mesmas são ordenadas por meio do grau de similaridade (do maior para o menor).

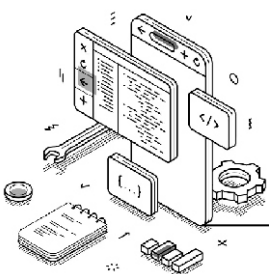


Figura 46: Resultados possíveis soluções para o problema.

Possíveis Soluções para o Problema

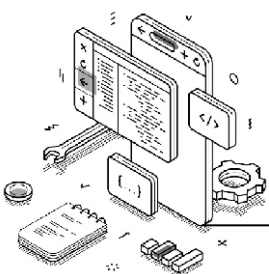
Código	Problema	Solução	Similaridade
11	Nível de sinal óptico acima do permitido para o funcionamento da ONU	-Verificar o estado do conector óptico, se não está danificado ou com sujeira; -Efetuar a limpeza do conector óptico; -Verificar se o cabo óptico não possui curvaturas acentuadas, desde a ONU até a caixa NAP; -Refazer o conector óptico; -Testar o nível de sinal óptico na caixa NAP; se estiver dentro do nível aceitável refazer o conector óptico do cabo do cliente; -Se não resolver com todos os passos descritos anteriormente o problema está no cabo óptico do cliente, sendo necessário realizar a substituição do mesmo.	100%
12	Nível de sinal óptico acima do permitido para o funcionamento da ONU	-Verificar o estado do conector óptico, se não está danificado ou com sujeira; -Efetuar a limpeza do conector óptico; -Verificar se não possui curva acentuada dentro da PTO; -Verificar se o cabo óptico não possui curvaturas acentuadas, desde a ONU até a caixa NAP; -Refazer a fusão do cordão óptico da PTO; -Comparar o nível de sinal óptico com outros clientes da mesma caixa NAP; -Se mais cliente estiverem com o nível de sinal óptico acima do aceitável, o problema está na Caixa NAP; -Fazer uma fusão com um cordão óptico na porta do splitter, testar o nível de sinal óptico e comparar o nível de sinal óptico entre a ONU e a caixa NAP, se for um nível de sinal bem próximo entre a ONU e o NAP o problema pode estar na porta do splitter; -Se a diferença de sinal óptico entre a ONU e o NAP, refazer a fusão; -Se não resolver com todos os passos descritos anteriormente o problema está no cabo óptico do cliente, sendo necessário realizar a substituição do mesmo.	50%
10	ONU com problema na porta óptica	Efetuar limpeza do conector óptico e na porta óptica da ONU, se não lancar o LED LOS e a OLT não identificar a ONU efetuar a troca da ONU, pois a mesma está com problema na porta óptica.	50%

Fonte: Ferrari *et al.* (2019).

Após o desenvolvimento do protótipo foi realizada uma série de testes com todas as funcionalidades propostas pela ferramenta, visando detectar falhas e inconsistências. Após estes testes, a ferramenta foi disponibilizada para os profissionais dos departamentos de administração de redes e *call center*, pois esses departamentos trabalham internamente dentro do provedor e fazem interação direta ou indiretamente com os clientes e os profissionais que trabalham na rede FTTx no ambiente externo, ajudando a realizar atividades diárias e auxiliando-os na detecção de problemas e na solução dos mesmos, que dizem respeito à rede FTTx. Durante a realização dos testes fez-se um acompanhamento diário, a fim de verificar possíveis problemas e falhas na ferramenta, para que fossem realizadas correções, ajustes necessários e, também realizar uma validação da ferramenta proposta a fim de verificar as impressões dos usuários sobre sua utilização.

Conforme apresentado nesta seção, um protótipo da ferramenta foi implementado e validado, permitindo o cadastro de todos os equipamentos (bem como sua localização) da rede, além da estruturação e implementação da técnica de RBC para buscar os casos mais semelhantes aos problemas apresentados à ferramenta.

A aquisição do conhecimento para elaboração da base de casos foi uma das tarefas mais difíceis, pois para a sua realização foi necessário conversar, informalmente, com diversos profissionais de departamentos diferentes, sendo que nem sempre os mesmos possuíam tempo livre suficiente para efetuar as trocas de informações e recolhimento das informações mais relevantes e necessárias. Além da aquisição do conhecimento, outra dificuldade enfrentada envolveu a definição das informações necessárias para compor a base de casos, de maneira que fosse estabelecido um padrão de fácil entendimento, facilitando a inserção de novos casos e localização de soluções para novos problemas relatados. A possibilidade de inserção de novos casos, de forma interativa pela ferramenta, é um requisito importante, pois dá

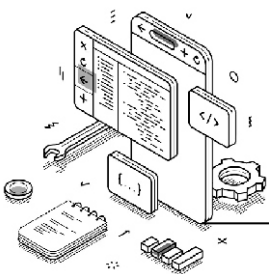


liberdade aos especialistas do domínio (no caso, as redes FTTx) para atualizarem a base de casos, incluindo novos casos ou alterando atributos de casos já existentes.

Para uma melhor utilização da ferramenta e das suas funcionalidades os autores realizaram um treinamento com os profissionais que utilizam esta ferramenta.

Para futuros trabalhos podem ser criadas funcionalidades capazes de armazenar, além dos casos, as seguintes informações: 1) quantidade de consultas realizadas para buscar soluções; 2) quantidade de problemas resolvidos; 3) quantidade de casos de novos; 4) quantidade de problemas não solucionados (e o motivo); 5) os problemas que são enfrentados com mais frequência; 6) as características mais utilizadas para localizar a solução de problemas. Essas informações possibilitarão a geração de relatórios e gráficos que podem auxiliar na supervisão do trabalho desenvolvido no provedor de Internet. Além disso, podem ser inseridas novas funcionalidades que permitam, por meio dos cadastros dos equipamentos e suas interligações, gerar um esquema de ligação por meio de imagens com diagrama de interligação, facilitando ainda mais a compreensão do dimensionamento dos equipamentos e suas interligações. Podemos, também, criar funcionalidades relacionadas aos níveis de acesso dos usuários ao sistema, fazendo com que os técnicos experientes (com conhecimento aprofundado) tenham permissão de administrador, podendo cadastrar e alterar as informações no sistema e, os técnicos com pouco conhecimento, tenham somente a permissão de efetuar consultas e localizar soluções, assim evitando problemas de cadastro de informações erradas ou incompletas.

Outra sugestão para trabalhos futuros consiste em disponibilizar a ferramenta de modo *off-line* (sistema instalado nos *notebooks* dos técnicos). Assim, quando os técnicos realizarem serviços externos, eles poderão utilizar o sistema sem necessidade de ligar para os técnicos do *call-center* e/ou acessarem a Internet, podendo realizar consultas de informações e buscar soluções para os problemas enfrentados diariamente. Neste caso, diariamente ou semanalmente a base de dados seria atualizada com a base de dados do servidor.





EXPERIÊNCIAS DESENVOLVIDAS COM REDES NEURAIS ARTIFICIAIS

A

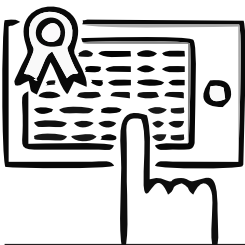
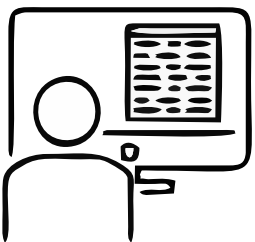
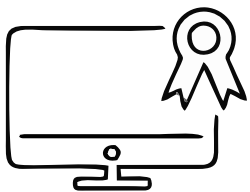
s RNAs (Redes Neurais Artificiais) são um paradigma de processamento de informação inspirado no sistema neural biológico. As RNAs são sistemas massivamente paralelos e distribuídos, formados por unidades de processamento simples, que calculam determinadas funções, normalmente não lineares. Essas unidades são distribuídas em camadas, sendo interligadas por conexões, as quais se associam a pesos.

Estes pesos armazenam o conhecimento representado na rede, servindo para ponderar as entradas recebidas por unidade constituinte (Haykin, 2007; Vieira, Parreira & Silveira, 2014; Basso *et al.*, 2016).

Uma das características mais importantes de uma RNA é a capacidade de aprender por meio de exemplos, denominados de conjunto de treinamento, sendo estes apresentados a ela durante a etapa de treinamento. Durante o aprendizado, a rede consegue extrair os padrões existentes neste conjunto de dados e, após finalizar a etapa de treinamento, ela aprende o suficiente para generalizar o problema. Ao final, na etapa de teste, a RNA consegue produzir respostas satisfatórias para dados desconhecidos, que fazem parte do mesmo problema, para o qual ela foi treinada (Haykin, 2007).

Segundo Palma Neto e Nicoletti (2005), os algoritmos de treinamento das RNAs podem ser agrupados em supervisionados e não-supervisionados. Nos supervisionados, além das informações, cada exemplo de treinamento traz a classe a qual ele pertence. Nos grupos não supervisionados a RNA é treinada sem considerar as informações da classe a qual cada exemplo está associado.

Existem várias arquiteturas para a implementação de uma RNA, mas estas podem ser divididas, basicamente, quanto ao número de camadas, e pelos tipos de conexões entre os neurônios (Fausett, 1994). De acordo com o número de camadas, a arquitetura de uma RNA pode ser



denominada de camada única ou de múltiplas camadas. As RNAs com múltiplas camadas distinguem-se de redes com camada única pela presença de camadas ocultas. Além disso, são utilizadas em problemas mais complexos, que necessitam analisar várias combinações das entradas da rede, como por exemplo, o diagnóstico de diabetes (exemplo que será visto na seção 6.1).

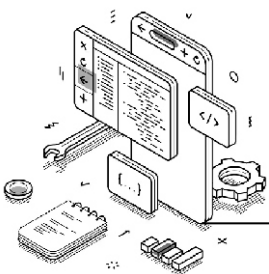
Quanto aos tipos de conexões, que definem a direção do fluxo de ativações, as RNAs são definidas como *feedforward* e *feedback*. As RNAs *feedforward*, também conhecidas como acíclicas ou não recorrentes, são estruturadas em camadas, e não possuem realimentação de suas saídas para as entradas. Estas podem ter de uma ou mais camadas em sua configuração. Dentre os algoritmos para treinamento de redes *feedforward*, o *back-propagation* está entre os que proporcionam uma maior eficiência. Já as RNAs *feedback*, também denominadas de cíclicas ou recorrentes, diferenciam-se das não recorrentes por terem em sua estrutura a realimentação das saídas para as entradas (Haykin, 2007).

Para que uma rede neural tenha sucesso na resolução de problemas práticos, é necessário que ela consiga representar funções lineares e não-lineares, assim como o algoritmo de treinamento deve ser suficientemente robusto e genérico, não restringindo a eficácia na resolução de problemas. As redes neurais de múltiplas camadas, com o algoritmo para treinamento *back-propagation*, contemplam essas características (Massad *et al.*, 2004).

As Redes Neurais Artificiais (RNAs) são algoritmos utilizados no processamento de informação, que tentam imitar e/ou simular o modelo de sistema neural biológico humano. Essas redes são sistemas que se apresentam de forma paralela e distribuída, e são formados por unidades de processamento mais simples, chamados neurônios, que calculam determinadas funções, geralmente não lineares. Esses neurônios situam-se em camadas, sendo interligadas por conexões, as quais são associadas a pesos. Estes pesos armazenam o conhecimento representado na rede, servindo para fazer um balanço entre as entradas recebidas por unidade constituinte da rede (Haykin, 2007).

Uma RNA possui a capacidade de aprender por meio de exemplos, que são conhecidos por conjuntos de treinamento, sendo estes dados aplicados durante a fase de treinamento da rede. Mediante a fase de treinamento, a rede armazena um padrão de aprendizado capaz de generalizar um determinado problema. Finalizando a aplicação da RNA, passa-se por uma etapa de teste, cuja mesma utiliza dados dos pesos armazenados na etapa de treinamento, e então cruza determinados dados produzindo respostas satisfatórias para dados ainda desconhecidos, porém que fazem parte de um mesmo problema para o qual a rede foi treinada (Haykin, 2007).

Além de entender o funcionamento de uma RNA, também é importante estudar sua arquitetura, que é o que determina a capacidade de processamento da rede. Escolher corretamente o número de conexões pode ser um passo decisivo para que o processo de treinamento seja bem sucedido. As redes podem ser classificadas em recorrentes ou não recorrentes (Velasco, 2007).



As RNAs recorrentes, ou redes do tipo *feedback*, possuem realimentação das saídas para as entradas, tendo como saída os valores determinados por entradas atuais e pelas saídas anteriores. Sua estrutura não é obrigatoriamente organizada em camadas; essas redes também podem apresentar interligações entre neurônios da mesma camada e entre camadas não subsequentes (Velasco, 2007).

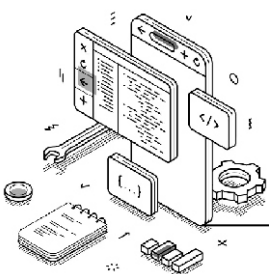
As redes não recorrentes, ou do inglês *feedforward*, são aquelas que não possuem realimentação de suas saídas para as suas entradas, por isso, são conhecidas como “sem memória”. Esse tipo de rede pode ser de camada única ou pode ser formada por múltiplas camadas. Essa arquitetura de redes em camadas é formada por um conjunto de neurônios de entrada, uma camada de saída e uma ou mais camadas intermediárias ou ocultas. Nas redes não recorrentes, não existem conexões ligando um neurônio de uma camada a outro neurônio de uma camada anterior nem a um neurônio de uma mesma camada. Esse tipo de rede *feedforward* é o mais utilizado atualmente, chamando a atenção para a distinção para as suas saídas que são sinal funcional e sinal de erro (Velasco, 2007).

Os sinais funcionais são sinais de entrada que se propagam para frente (neurônio a neurônio) através das camadas da rede e terminam na saída da rede como um sinal de saída. São chamados assim porque em cada neurônio da rede pelo qual o sinal passa, esse sinal é calculado como uma função das entradas pelos pesos associados àquele neurônio. Já os sinais de erro originam-se nos neurônios de saída e se propagam para trás (camada por camada) através da rede (Velasco, 2007).

Após os sinais de entrada terem se propagado por todas as camadas da rede, chegamos ao final de uma iteração, onde ocorre a atualização/ajuste dos pesos, que contém o conhecimento da rede. Segundo Tissot, Camargo & Pozo (2012), quando trabalhamos com os pesos a atenção deve ser voltada para a inicialização e ajuste dos mesmos, sendo esse um processo fundamental para a convergência e para evitar *overfitting* (excesso de ajuste nos pesos), problema que geralmente acontece em padrões de treinamento. Ainda com relação ao ajuste dos valores dos pesos na rede, existem vários algoritmos já consagrados, tais como o *perceptron*.

O *perceptron* foi desenvolvido por Frank Rosenblatt por volta do ano de 1958. Este algoritmo usa uma forma simples de aprendizado supervisionado, baseando-se na modificação dos pesos de modo a reduzir a taxa de erro da rede neural. O *perceptron* é composto pelas camadas de entrada e de saída, possuindo ainda um componente conhecido como nível de ativação. A camada de entrada e o nível de ativação do *perceptron* trabalham com os valores de -1 representando a não ocorrência do fator analisado e de +1 quando há a ocorrência do padrão a ser analisado (Luger, 2013).

O conhecimento gerado pelo algoritmo é armazenado em pesos, que são representados por conjuntos numéricos do tipo real. Os níveis de ativação do *perceptron* são obtidos pela soma dos valores ponderados das entradas. Depois de obtidos estes valores, os mesmos passam por uma função limiar que determina o valor de sua saída, que quando ficar próximo de +1 representa



forte ocorrência de acerto, ou pelo contrário -1, que representa baixa ocorrência de acerto (Luger, 2013).

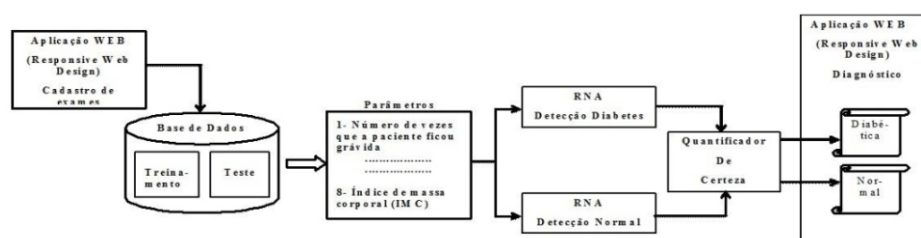
Este algoritmo trabalha sobre a premissa de que os padrões de entrada sejam necessariamente padrões lineares separáveis, ou seja, conjuntos que possam ser separados por uma ou mais características (Haykin, 2001). A rede neural *perceptron* está classificada como sendo uma rede *feedforward*, cujo fluxo de informações em sua estrutura reside sempre no sentido da camada de entrada para a camada de saída, não existindo qualquer tipo de realimentação dos valores produzidos para a sua camada de entrada (Silva *et al.*, 2010).

Considerando a arquitetura de uma rede neural do tipo *perceptron*, como mencionado anteriormente, devido às suas características estruturais, as funções de ativação utilizadas são as funções degrau, ou seja, possuem apenas duas possibilidades de saída (Silva *et al.*, 2010).

6.1 Sistema de Auxílio ao Diagnóstico de Diabetes (Vieira, Parreira & Silveira, 2016)

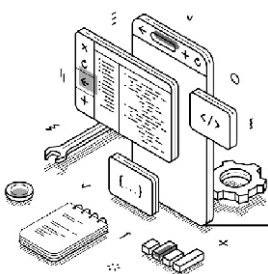
A solução apresentada nesta seção envolve a implementação de um Sistema de Informação para o apoio ao diagnóstico de diabetes, empregando RNAs, o SADD (Sistema de Auxílio ao Diagnóstico de Diabetes), desenvolvido por Vieira, Parreira & Silveira (2016). Inicialmente uma aplicação *web* foi desenvolvida para que os usuários (profissionais da área de saúde) pudessem interagir com o sistema. Esta aplicação é responsável por receber os dados dos pacientes e submetê-los à RNA para que o teste seja realizado. A escolha de uma aplicação *web* deveu-se ao fato de que a mesma permite a padronização das informações a serem salvas e, também, pela mobilidade, pois a mesma é responsiva, podendo ser acessada por meio de qualquer *smartphone*, *tablet* ou computador *desktop*. Essa mesma aplicação é usada, posteriormente, para que os profissionais da área de saúde realizem a inserção dos dados pessoais e obtenham apoio para realizarem o diagnóstico de diabetes.

Figura 47: Arquitetura do SADD.



Fonte: Vieira, Parreira & Silveira (2016).

A Figura 47 apresenta a arquitetura do SADD. Nesta figura também é possível observar o fluxo de dados do sistema. A aplicação *web* é responsável por receber os parâmetros dos pacientes e enviá-los para a base de dados, onde os mesmos ficam armazenados para futuros treinamentos das RNAs. Esses dados



são utilizados pelas RNAs para que seja realizado o teste da amostra. Logo após esse teste, as RNAs enviam os resultados obtidos para um quantificador de certeza, que faz a análise entre os resultados obtidos por meio das RNAs, devolvendo o resultado, ou seja, a mensagem contendo o pré-diagnóstico.

Os parâmetros são enviados para as RNAs para que as mesmas realizem a análise das informações e as enviem para o quantificador de certeza, responsável por realizar a análise entre as duas redes e definir a porcentagem de certeza do diagnóstico. Após realizada a quantificação da amostra, o resultado é enviado para a aplicação *web* em forma de uma mensagem “DIABÉTICO % DE CERTEZA” ou “NÃO DIABÉTICO % DE CERTEZA”.

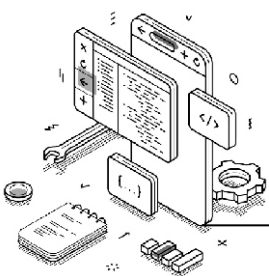
O uso de RNAs foi escolhido devido a sua habilidade de adaptação e aprendizagem, já que podem lidar com dados imprecisos e situações não totalmente definidas. Uma rede treinada corretamente tem a habilidade de generalizar quando são apresentadas as entradas que não estão presentes em dados já conhecidos pela mesma. Isso pode acontecer com dados relacionados à diabetes, pois não há uma uniformidade dos dados a serem analisados (Haykin, 2007).

Estudos realizados apontam que as RNAs têm um grau maior de acerto no diagnóstico de diabetes. O trabalho apresentado por Sujatha (2014) realizou uma comparação entre RNAs e regressão matemática, atingindo um índice de 17% a mais de acertos usando RNA. Esse teste foi realizado utilizando a mesma base de dados que será usada pela aplicação aqui apresentada, a PID.

Na base de dados do SADD são armazenadas todas as informações que o levantamento coletar, sendo que a mesma é dividida em 2 partes: uma para fazer o treinamento das duas RNAs e a segunda para fazer o teste das mesmas. Essa divisão é de 30% da base para o teste e 70% para o treinamento (Kayaer & Yildirim, 2003).

Foram desenvolvidas duas RNAs, sendo que as duas foram treinadas com as mesmas informações. Entretanto, uma das RNAs realiza a detecção da diabetes e a outra busca os casos normais, onde não é encontrada a doença. Cada uma das RNAs fornece uma porcentagem de certeza que é calculada pelo algoritmo da RNA, por meio da implementação de um quantificador de certeza. O quantificador de certeza tem, como objetivo, a partir das porcentagens geradas pelas duas RNAs, quantificar uma porcentagem geral, classificando se a amostra treinada é ou não diabética. Esse resultado é devolvido para a aplicação *web* que, por sua vez, mostra uma mensagem informando se esse paciente é ou não diabético e a porcentagem de certeza que o sistema conseguiu atingir.

Para treinar e testar a aplicação da RNA no sistema desenvolvido foi utilizada a base de dados da Universidade da Califórnia (UCI, 2010), denominada *Pima Indians Diabetes* (PID). As informações para compor esta base foram coletadas na comunidade indígena *Pima*, que vive perto de Phoenix, Arizona, Estados Unidos. Todas as pessoas, cujos dados foram coletados, são mulheres, com idade igual ou superior a 21 anos.



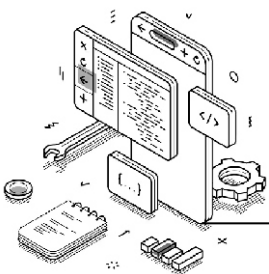
Diversas pesquisas foram realizadas nesta comunidade, pois ela é conhecida por apresentar a maior taxa de incidência de diabetes do mundo. Entre os adultos, 50% são diabéticos, além de apresentarem uma alta prevalência de obesidade (Baier & Hanson, 2004). Devido a estas características, os pesquisadores resolveram criar a base de dados PID, que é composta por 768 pessoas, onde 268 destas foram classificadas como tendo diabetes, do tipo 2, o que resulta em 34,9% da base. Os outros 65,1% não possuem diabetes e são classificados como normais. Para cada pessoa que compõe a base, é informado: o número de vezes que a paciente ficou grávida, teste oral de tolerância à glicose, pressão arterial diastólica (mm Hg), espessura da dobra cutânea do tríceps (mm), idade (anos), índice de massa corporal (IMC) e a classificação (possui ou não diabetes).

Os principais requisitos que foram considerados na implementação do SADD foram:

- Importação automática das informações contidas na base *Pima*: a base de dados PID está disponível na *web*, no formato TXT (texto), portanto, foi preciso importá-la para a base de dados *MySQL* do SADD, sendo que os dados nela contidos devem manter a sua integridade;
- Treinamento e teste das RNAs: O administrador do sistema deve treinar as redes neurais sempre que houver alterações significativas nos parâmetros e/ou nas amostras referentes a diabetes. O treinamento deve se dar com os 70% da base de dados *pima*, usando cerca de 538 amostras para o treinamento, os outros 30 da amostra são utilizados para o teste das redes neurais, onde são usadas 230 amostras;
- Teste das amostras: após o treinamento das RNAs, as mesmas devem passar por um teste, para que se tenha a certeza de que o conhecimento adquirido pelas mesmas é o suficiente para realizar um diagnóstico seguro da doença;
- Cadastro das informações via interface *web*: as informações dos usuários são cadastradas por meio de uma aplicação *web*; a mesma deve conter um formulário, onde são informados os parâmetros sobre a doença; logo após o usuário entrar com os parâmetros para teste, a aplicação deve submeter essa amostra ao teste e devolver o resultado para o usuário novamente.

Para desenvolver esse projeto foram utilizadas tecnologias *open source*, disponíveis gratuitamente, podendo-se destacar o HTML5. Para o desenvolvimento das RNAs utilizou-se a linguagem de programação Java, que proporciona uma performance estável e, também, pela facilidade na implementação de RNAs com esta linguagem, já que a linguagem Java possui instruções que facilitam processamento de cálculos, que é a base das RNAs (Gonçalves, 2006). Para a base de dados utilizou-se o SGBD *MySQL*, que é muito utilizado em aplicações *web* de pequeno porte.

Como descrito anteriormente, a base de dados *Pima* é disponibilizada no *site* da Universidade da Califórnia, em formato de arquivo do tipo texto (TXT). Para facilitar o desenvolvimento da aplicação e possibilitar que o treinamento das

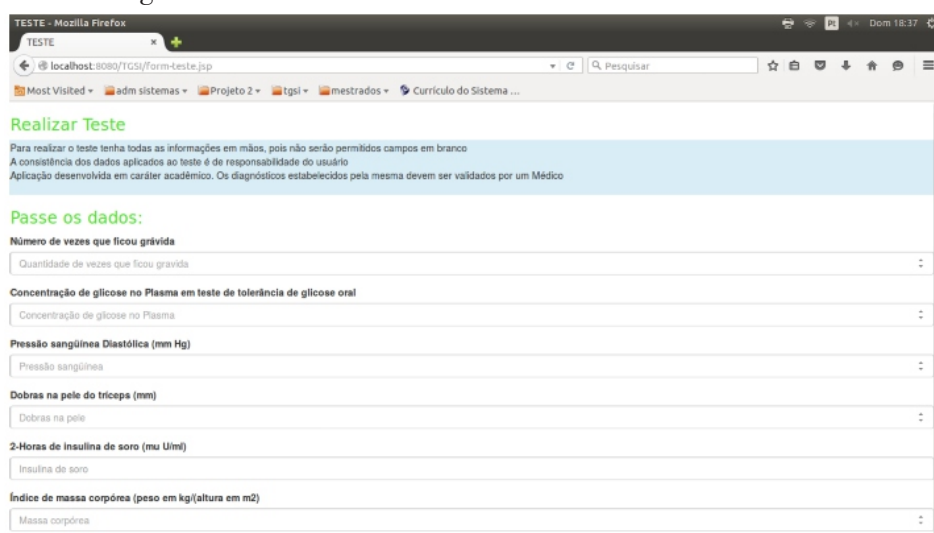


RNAs fosse realizado de forma mais rápida, esta base foi transferida para uma base de dados *MySQL*, de forma automatizada, por meio de um aplicativo desenvolvido em linguagem de programação Java. Estes dados foram utilizados no treinamento e testes das RNAs.

Por meio da página principal do SADD, os usuários (administrador e usuário final) dispõem de acesso às funcionalidades do sistema. Por meio do menu “SADD” os usuários têm acesso às funcionalidades do sistema, que são: Cadastro de novos exames, gerar gráfico das redes neurais e uma página *Sobre*, com informações sobre o sistema.

Na Figura 48 pode-se observar onde é realizado o cadastro das amostras de dados dos pacientes, por meio de um formulário. Esse formulário conta com um botão na parte inferior da tela onde os dados são validados e, se estiverem dentro do padrão de entrada, são submetidos às redes neurais para serem avaliados, bem como são armazenados na base de dados.

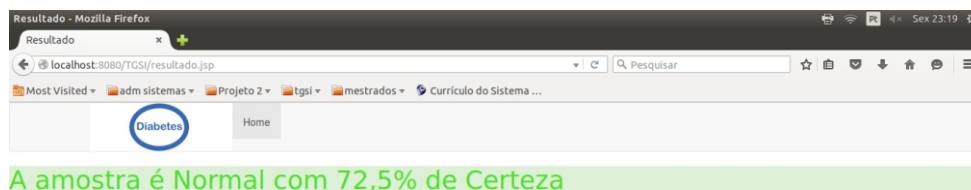
Figura 48: Cadastro das Amostras de Dados dos Pacientes.



Fonte: Vieira, Parreira & Silveira (2016).

Ao submeter esses dados, o SADD retorna a mensagem mostrando o diagnóstico que o sistema encontrou, com base nos resultados das RNAs, como mostra a Figura 49.

Figura 49: Diagnóstico.

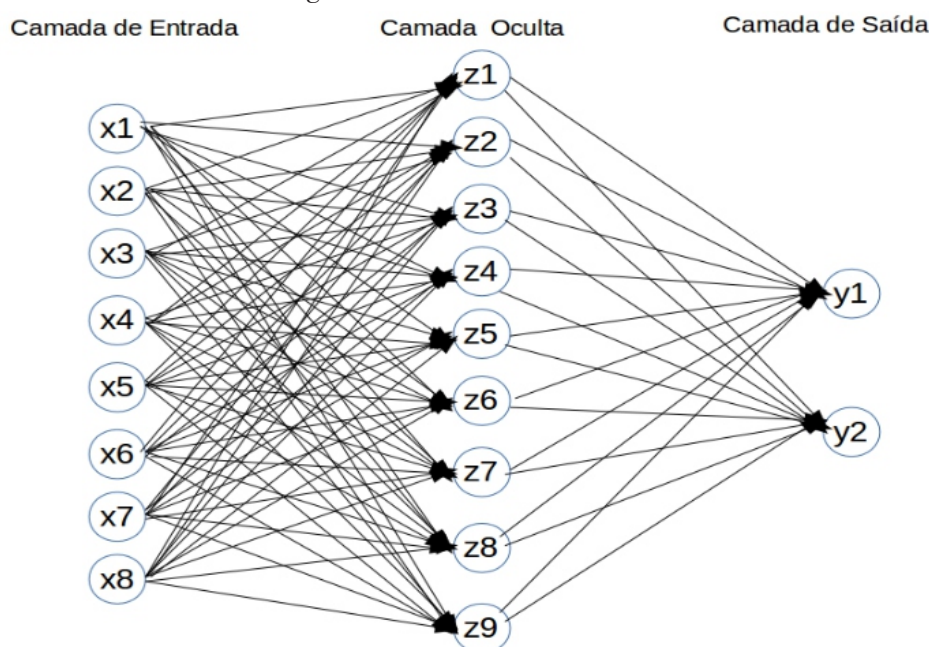


Fonte: Vieira, Parreira & Silveira (2016).

O resultado final mostra a classificação das amostras submetidas ao SADD. A mensagem mostra a porcentagem de certeza obtida por meio de uma média dos pesos dos neurônios da camada de saída e uma classificação atribuída pelo quantificador de certeza: Normal, Diabética ou Indefinida. No caso da amostra ser Normal, ou seja, o paciente não possuir a doença, a mensagem será mostrada na cor verde; caso o quantificador a classifique como indefinida, identificando que não foi possível obter um resultado, a mensagem será mostrada na cor laranja e, se a amostra for classificada como diabética, a mensagem mostrada será na cor vermelha.

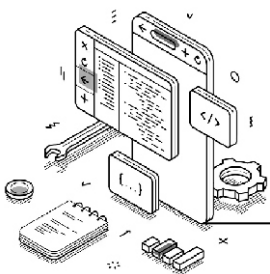
O SADD possui uma funcionalidade que permite que o administrador do sistema limpe os dados de um treinamento anterior e realize novamente o treinamento das redes. Quando o treinamento das redes é disparado, ocorre de forma paralela entre as RNA1 (para diagnóstico de casos diabéticos) e a RNA2, que tem por objetivo descobrir os casos normais. Para esse treinamento os dados são submetidos a um algoritmo de normalização, onde os mesmos são dispostos em uma faixa de -1×10^6 a 1×10^6 para serem submetidos às RNAs. Essa faixa de valor é utilizada para que não haja um intervalo muito grande entre os dados passados para as RNAs. Após a normalização, os dados são passados para as RNAs. Cada uma das RNAs possui três camadas: entrada, oculta e de saída, como mostra a Figura 50.

Figura 50: Camadas das RNAs.



Fonte: Vieira, Parreira & Silveira (2016).

Como se pode ver na Figura 50, existem 8 neurônios na camada de entrada, correspondentes aos parâmetros que são nativos da base utilizada (base *Pima*), parâmetros estes que são passados para a camada oculta. Na camada oculta existem 9 neurônios que são treinados pelas amostras, por meio dos parâmetros



passados pela camada anterior. Esses neurônios recebem uma espécie de peso que é salvo no banco de dados para posteriormente serem utilizados nos testes. Por fim, a camada de saída possui dois neurônios (que indicam os possíveis resultados), cujos pesos também são salvos na base de dados.

Nos testes das RNAs, tanto utilizando a base de dados nativa como na entrada de novas amostras, os parâmetros de teste são submetidos à normalização. Após a normalização os dados atribuídos aos neurônios são buscados na base de dados e é realizado o teste da amostra, teste esse que atribui quatro valores, sendo dois para cada RNA.

O quantificador de certeza recebe os dados das saídas das RNAs e, por meio de seu algoritmo, analisa os quatro valores que são obtidos por meio do teste ao que a amostra foi submetida e aplica métricas para que seja calculada a saída final que será submetida para a aplicação *web* e, então, mostrada para o usuário que realizou o teste o resultado.

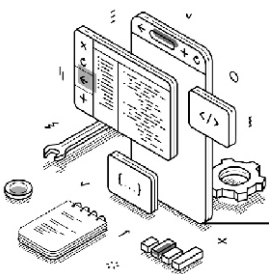
A aplicação foi testada e validada por meio da base de dados PID, utilizando um fator de certeza de no mínimo 75% (Lorenzi & Silveira, 2011). A base PID já foi testada e validada por um número expressivo de pessoas especializadas na área de diabetes (UCI, 2010). Para os testes realizados pelo SADD, utilizando os dados da base PID, obteve-se um resultado satisfatório após 83 treinamentos, já que o índice de acerto ficou dentro dos parâmetros esperados.

Neste contexto, este trabalho teve como principal contribuição o desenvolvimento da ferramenta SADD, que pode ser utilizada por profissionais da área da saúde no auxílio ao diagnóstico da diabetes tipo 2. Como trabalhos futuros os autores destacam a construção de uma base de dados da população brasileira permitindo, assim, que o SADD use parâmetros mais próximos da realidade brasileira. Outra possibilidade é utilizar outros parâmetros, tais como a porcentagem de agrotóxicos aplicados aos alimentos.

6.2 Aprendendo com o Zag: Jogo Educacional Digital para Alfabetização utilizando Redes Neurais Artificiais (Basso et al., 2016)

O jogo *Aprendendo com o Zag* foi desenvolvido por Basso *et al.* (2016), com o intuito de auxiliar alunos do Ensino Fundamental, com idade entre cinco e sete anos, no seu processo de alfabetização. O projeto envolveu o desenvolvimento de um jogo educacional digital codificado utilizando tecnologias *web* tais como HTML5, CSS3 e *Javascript*. O jogo dispõe de algumas fases para que a criança possa aprender a escrever e identificar corretamente as vogais do alfabeto. Nesta seção vamos abordar os aspectos que compreendem a utilização de RNAs em uma das fases do referido jogo.

A aplicação de RNAs compõe um ponto em destaque no jogo, já que o mesmo conta com um módulo onde são explorados todos os recursos do algoritmo *perceptron*, com a finalidade de validar as respostas dos jogadores de maneira correta e precisa, gerando resultados condizentes com as respostas



fornecidas pelo jogador. Basicamente este módulo é utilizado nas fases do terceiro nível do protótipo do jogo implementado, onde o jogador deve escrever as vogais na tela.

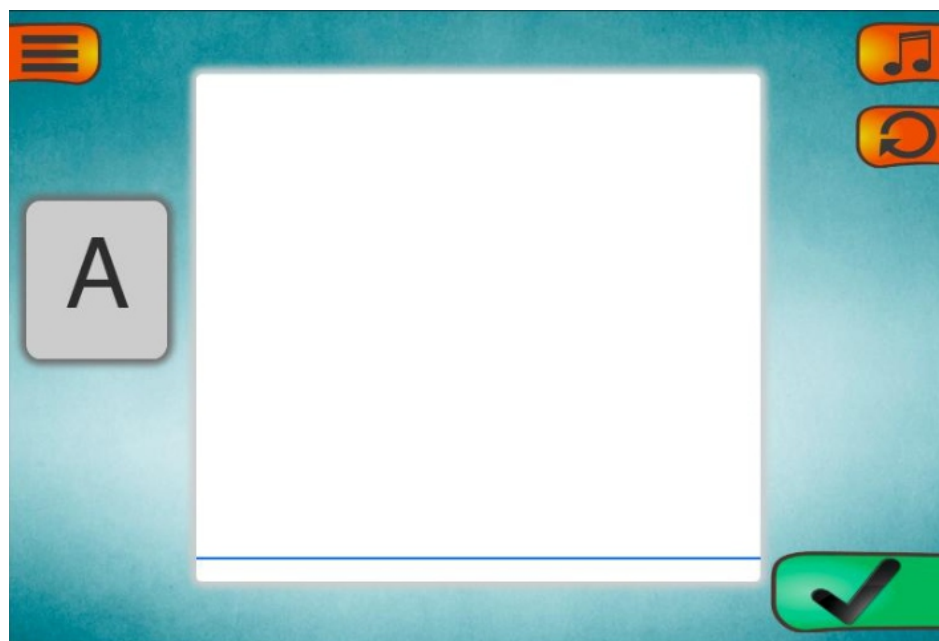
No último nível do jogo (terceiro nível), *Zag* (que é o personagem central do jogo) encara seu primeiro dia de prova na escola. Neste nível o aluno deve escrever as vogais do alfabeto no quadro branco, porém agora sem o auxílio da letra em vermelho (como ocorria nas fases anteriores) (Basso *et al.*, 2016), aumentando consideravelmente o nível de dificuldade do nível. Neste nível as respostas são validadas por uma RNA, que realiza a comparação dos dados de

treinamento, que são as respostas desejadas para as fases, com a resposta do jogador, definindo se a resposta do jogador está certa ou errada, mostrando o resultado ao jogador.

O importante a ressaltar nesta etapa é que, a base de treinamento disponível para a fase de teste da RNA já se encontra armazenada no protótipo, não necessitando o treinamento a cada execução. Além disso, os dados contidos nesta base foram obtidos por meio de tentativa e erro, onde o conjunto de dados que se mostrou mais relevante perante os resultados desejados, foi utilizado para compor a base de treinamento.

Nesta fase do jogo a professora de Zag diz uma letra ao jogador que deve escrevê-la no quadro em branco, sem auxílio de nenhuma imagem, e clicar no botão verde para verificar a sua resposta. A Figura 51 apresenta a interface do terceiro nível. A resposta (escrita da letra) nesta fase é avaliada por uma RNA baseada em padrões.

Figura 51: Terceiro Nível do Jogo Aprendendo com o Zag.



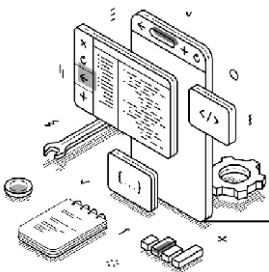
Fonte: Basso *et al.*, 2016.

Para tanto, precisou-se desenvolver uma classe chamada de PIC (Processamento de Imagens em *Canvas*) que gera os padrões de treinamento e de teste da RNA, para que se possa ter uma unificação no padrão de coleta dos dados com os quais a RNA irá trabalhar. A classe PIC possui dois *canvas* auxiliares ocultos (não são exibidos ao jogador). O primeiro passo da PIC é fazer o recorte do desenho feito pelo jogador, eliminando os espaços em branco que são desnecessários e duplicar a imagem para o primeiro *canvas* auxiliar que recebe o nome de *canvasRecorte*. Esse processo de redimensionamento é feito pelo cálculo das dimensões *width* (largura), *height* (altura) e também as posições *x* e *y* do plano cartesiano, no conjunto de dados obtidos pelo método *getImageData()* do contexto do *canvas*. Após ter recuperado no *canvas* de recorte a parte relevante da imagem, recupera-se novamente, por meio do método *getImageData()* a matriz de dados atualizada.

A partir desta atualização, utiliza-se a função *putImageData()*, que foi reescrita para atender as necessidades do projeto, para desenhar os dados no segundo *canvas* auxiliar que possui um tamanho padrão de sete *pixels* de largura por nove de altura. Então se recupera novamente a matriz de dados que terá exatamente as dimensões sete por nove e gera-se a matriz *pixelada* (matriz que possui conjuntos de valores em que cada posição da mesma representa um pixel da tela), atribuindo-se “1” para a posição da matriz onde o *pixel* possuir cor e “-1” para onde o *pixel* for branco ou não possuir cor. Então, após ter realizado todo o processamento descrito, obtêm-se uma matriz *pixelada*, pelo método *getMatriz()* da classe PIC.

No jogo implementado, é importante ressaltar que a saída da classe PIC foi ajustada para uma matriz *pixelada* de sete colunas por nove linhas, por ser a menor matriz *pixelada* que pode representar uma letra do alfabeto com precisão. Esta medida foi obtida de forma empírica, na base de tentativa e erro, a partir da realização de vários testes, começando com medidas maiores para a matriz *pixelada*, chegando-se ao menor número possível da matriz em que não havia perda na taxa de reconhecimento. Outro ponto relevante é que uma matriz de proporção sete colunas por nove linhas representa um total de 63 posições ou mapeamentos de *pixels* do *canvas*. Sendo assim, cada uma destas posições corresponde a um neurônio de entrada na rede neural. Quanto maior a quantidade de informações processadas pela rede neural, mais lenta será sua execução, como desempenho é fundamental em um jogo, optou-se por adotar o menor conjunto de entradas.

Uma das vantagens da utilização da classe PIC para a recuperação dos dados de treinamento e teste da RNA é que, como o jogo é responsivo, ou seja, atende a várias resoluções de tela, a matriz *pixelada* no final do processo PIC sempre terá o mesmo tamanho, o que possibilita que se tenha somente um conjunto de pesos de treinamento gerados para a execução dos testes no RNA em diversas resoluções de tela. Então, com a classe PIC funcionando corretamente na interface do jogo, foram coletadas as matrizes *pixeladas* das vogais escritas por várias pessoas diferentes e, juntamente com a matriz *pixelada* de vogais digitais, foi criada a base de treinamento da RNA. Esta base de treinamento foi composta por 17 amostras para cada letra vogal, totalizando um total de 85 amostras, onde cada uma destas corresponde a uma saída da classe PIC. Estas

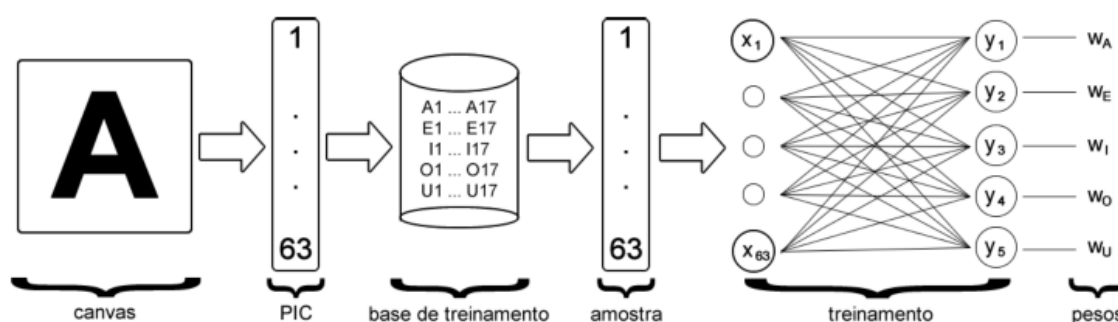


amostras foram geradas a partir da escrita de algumas pessoas que se dispuseram a contribuir com o projeto (17 voluntários). Não há um número correto de amostras para a RNA funcionar perfeitamente. Acreditamos que, quanto mais amostras existirem, melhor será o reconhecimento. Após a formação da base de treinamento, ela foi devidamente armazenada na classe *DadosRNA*, bem como os alvos do treinamento e do teste, e posteriormente os pesos gerados pelo treinamento da RNA.

Para a etapa de treinamento da RNA e geração dos pesos que contém o conhecimento da rede, foram executados basicamente seis passos de forma consecutiva. O primeiro deles é o passo 0, onde inicializamos a matriz de pesos. No próximo passo, o passo 1, definimos a condição de parada da rede neural, que é uma *flag* que recebe o valor *false*, e também é feita a inicialização do contador de passos necessários para o treinamento da RNA e de uma taxa de aprendizagem no valor de “0.005”, valor este obtido de forma empírica (tentativa e erro). Esta taxa ajuda a controlar a velocidade e a qualidade do conhecimento gerado pela RNA. No passo 2, foi implementada uma estrutura de repetição do tipo *for* que executa, para cada amostra de treinamento ($x_1 \dots x_n$) ($t_1 \dots t_n$), os passos 3, 4 e 5 do treinamento. Dentro deste laço de repetição, no passo 3 da etapa de treinamento, calculamos o potencial de ativação das saídas (Y_{in}) para cada neurônio de saída. Após isso, no passo 4, calcula-se a resposta do neurônio de saída (y), aplicando a função de ativação, que define o valor da saída (y). No passo 5, são atualizados os pesos se caso a saída (y) for diferente do valor do *target*/alvo (t). Executa-se esta estrutura de repetição até que a condição de parada seja verdadeira, então a função de treinamento retorna a matriz de pesos com o conhecimento da rede.

Para o jogo implementado foram necessários um total de 63 neurônios de entrada (x), já que utilizamos uma matriz 7×9 , e de 5 neurônios de saída (y), cada um representando a saída de uma vogal do alfabeto. Para o processo de treinamento foram necessários aproximadamente 23 passos de execução das instruções da RNA para obter o conhecimento necessário para a etapa de teste, os pesos (w). A Figura 52 representa as fases de execução da RNA, desde a coleta de dados da classe PIC até a finalização do treinamento da rede.

Figura 52: Coleta de Dados PIC e Treinamento da RNA.



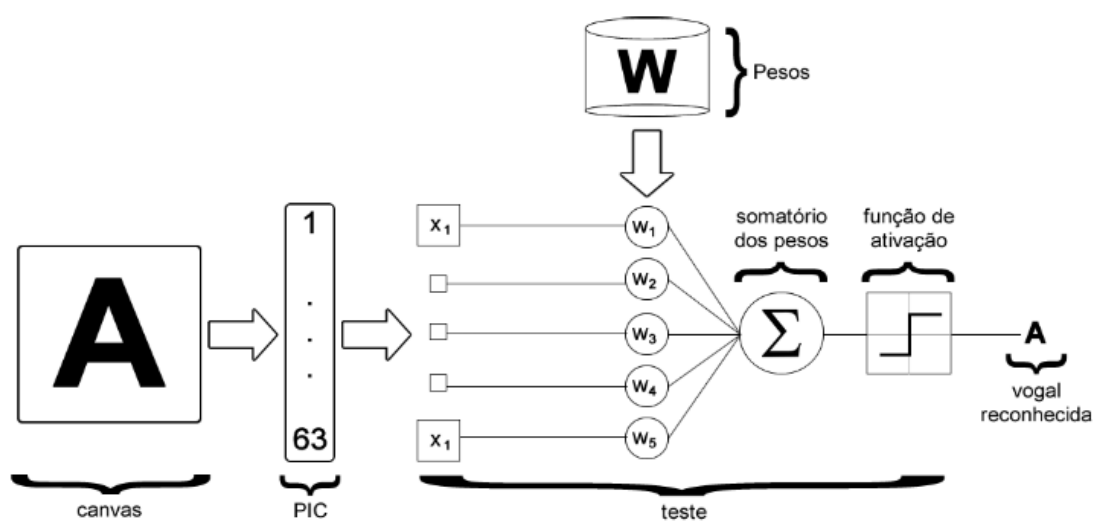
Fonte: Basso *et al.*, 2016.

Para evitar que a RNA tenha que executar a etapa de treinamento toda a vez que o botão verificar seja pressionado, os pesos resultantes do treinamento foram armazenados na classe de *DadosRNA* em um arquivo *Javascript*, para que possam ser recuperados rapidamente para a execução da etapa de teste. Neste mesmo nível, como mencionado anteriormente, toda a vez que o botão verde (verificar a resposta) for pressionado, a etapa de teste da RNA é executada. Para que isso aconteça, primeiramente os dados da base de treinamento são recuperados, e a classe PIC gera em tempo real a matriz *pixelada* que irá compor a base de teste da RNA, da mesma maneira que ocorre na etapa anterior ao treinamento da rede. Executados estes procedimentos, temos os dados necessários para a realização do teste.

Na etapa de teste, a RNA executa novamente os passos 3, 4 e 5 da rede com algumas adaptações. No passo 3, foi criada, novamente, a estrutura de repetição do tipo *for* que calcula o potencial de ativação da saída (Y_{in}) para cada neurônio de saída. Dentro desta estrutura de repetição, no passo 4, calculamos a resposta do neurônio de saída (y), aplicando a função de ativação obtendo, assim, a resposta do limiar da rede. Em uma RNA o limiar é o número central, que fica entre a resposta considerada correta e a incorreta. O limiar que foi definido para a RNA do jogo implementado foi o valor de “0.5”, valor obtido de forma empírica (tentativa e erro).

No passo 5 da RNA foi realizado o teste propriamente dito, por meio da comparação entre os dados do treinamento e os dados processados no teste. Então é feita a identificação da letra, se a letra solicitada for igual à letra identificada, a função de teste da RNA retorna verdadeiro, caso contrário retorna falso. A Figura 53 representa os procedimentos que ocorreram na etapa de teste da RNA.

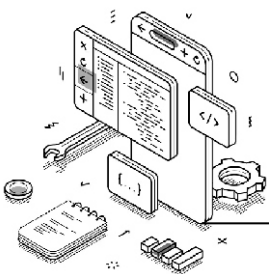
Figura 53: Coleta de Dados PIC e Teste da RNA.



Fonte: Basso *et al.*, 2016.

Quando a RNA identifica a letra, e esta for igual à letra solicitada, é disparado um sinal sonoro mostrando que o jogador acertou a letra. Posteriormente, é requisitado que o mesmo desenhe outra letra aleatória. Quando o jogador escrever corretamente todas as vogais e a RNA identificá-las, a fase é dada por encerrada.

Como trabalhos futuros, os autores pretendem desenvolver uma segunda versão do jogo educacional digital, incluindo todas as letras do alfabeto, além de disponibilizar outras fases, aplicando diferentes técnicas de IA. Outra opção seria melhorar a classe PIC para que a mesma possa criar padrões de identificação levando em consideração as cores e não somente a opção monocromática e, também levar em consideração a disposição da imagem, realizando ajustes de ângulo e distorção da mesma.





CONSIDERAÇÕES FINAIS

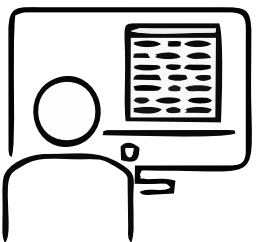
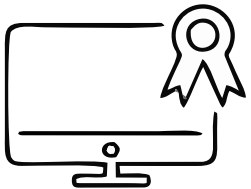
E

Este e-book apresentou alguns dos projetos desenvolvidos pelo grupo de pesquisa IATE-UFSM, por meio de protótipos implementados empregando diferentes técnicas de IA, tais como Sistemas Especialistas, Sistemas de Recomendação, Redes Neurais Artificiais e Raciocínio Baseado em Casos.

Todos os protótipos implementados, além de seguirem uma série de passos que compreendem desde o levantamento de requisitos até a modelagem, de acordo com os princípios da Engenharia de Software, também foram testados e validados com usuários reais, o que permitiu identificar possíveis melhorias e sugestões de trabalhos futuros.

Além dos projetos apresentados, o grupo de pesquisa também tem desenvolvido projetos nas áreas de Business Intelligence, por meio de Sistemas de Informação Empresariais empregando técnicas de IA, tais como o SIAD (Sistema Inteligente de Apoio à Decisão) como Ferramenta de Business Intelligence, elaborado por Frizon, Silveira & Cunha (2014). Outra área em estudo é a da Mineração de Dados, compreendendo o trabalho desenvolvido por Lengert & Silveira (2024), na área de Mineração de Dados climáticos.

O grupo de pesquisa também tem se dedicado ao estudo de ferramentas de IA generativa, aplicadas à Educação, tais como o chatGPT. Os estudos compreendem a aplicação de diferentes ferramentas de IA como apoio aos processos de ensino e de aprendizagem dos alunos dos Cursos de Bacharelado em Sistemas de Informação e de Licenciatura em Computação EaD, ambos ofertados pela UFSM (Universidade Federal de Santa Maria) – Campus Frederico Westphalen/RS.



BIBLIOGRAFIA



Experiências

de Desenvolvimento de Sistemas de Informação Inteligentes



REFERÊNCIAS

ABINPET - Associação Brasileira da Indústria de Produtos para Animais de Estimação. Mercado Pet Brasil 2022. 2022. Disponível em: https://abinpet.org.br/wp-content/uploads/2022/08/abinpet_folder_dados_mercado_2022_draft3_web.pdf. Acesso em: 08 out. 2022.

Almeida, L. R.; Silveira, S. R.; Cunha, G. B. Ambiental Web: Sistema Especialista para Apoio à Avaliação de Processos de Licenciamento Ambiental. In: EATI - Encontro Anual de Tecnologia da Informação, 2015, Frederico Westphalen /RS. VI EATI - VI Encontro Anual de Tecnologia da Informação. Frederico Westphalen /RS: Instituto Federal Farroupilha, 2015. Disponível em: <http://2015.eati.info/assets/anais/Longos/L1.pdf>. Acesso em: 30 abr. 2025.

Alvarez, E. B.; Siriani, A. L. R.; Vidotti, S. A. B. G.; Carvalho, A. M. G. Os Sistemas de Recomendação, Arquitetura da Informação e a Encontrabilidade da Informação. 2016. Disponível em: <https://www.scielo.br/j/tinf/a/YsgLRc86K3WZfcbXPQHq7Vg/?format=pdf&lang=pt>. Acesso em: 12 out. 2022.

American Psychiatric Association (APA). Manual de Diagnóstico e Estatístico de Transtornos Mentais (DSM-5, 2014). Porto Alegre: Artmed, 2014.

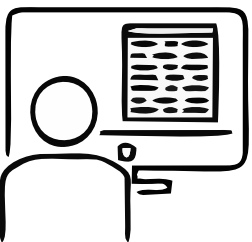
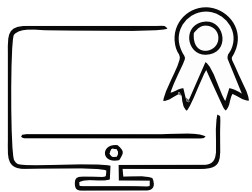
Anselmo, M. P. A.; Silveira, S. R. Previndex: Sistema Especialista para a área de Direito Previdenciário. Trabalho de Conclusão de Curso, Bacharelado em Sistemas de Informação, 2009. Porto Alegre: UniRitter. Acesso em: 22 abr 2017.

Araújo, W. J. et. al. Elementos Tecnológicos de Edição, Manipulação e Uso dos Livros Digitais. 2013. Disponível em: <http://www.periodicos.ufpb.br/index.php/ies/article/view/12969>. Acesso em: 04 mai. 2019.

Baier, L. J.; Hanson, R. L. Genetic studies of the etiology of type 2 diabetes in Pima Indians. Diabetes, 53, 1181–1186, 2004. Disponível em: diabetes.diabetesjournals.org/content/53/5/1181. Acesso em: 20 jan. 2015.

Bariani, I. C. D. Estilos cognitivos de universitários e iniciação científica. Campinas: UNICAMP, 1998. Tese de doutorado.

Barone, D. A. C. (Org). Sociedades Artificiais: A Nova Fronteira da Inteligência nas Máquinas. Porto Alegre: Bookman, 2003.



Basso, M.; Kliszcz, S.; Parreira, F. J.; Silveira, S. R. Jogo Educacional Digital para Auxílio à Alfabetização Utilizando Redes Neurais Artificiais. Revista Educaonline, v. 10, p. 1, 2016.

Bergmann, J. Aprendizagem Invertida para resolver o Problema do Dever de Casa. Porto Alegre: Penso, 2018.

Brasil. Lei 12.651/2012: Dispõe sobre a proteção da vegetação nativa. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato20112014/2012/lei/112651.htm. Acesso em: 15 abr. 2015.

Brusilovsky, P. Adaptive Hypermedia: An Attempt to Analyse and generalize 1995. Disponível em: <http://www.education.uts.edu.au/projects/ah/Brusilovsky.html>. Acesso em: 15 abr. 2016.

Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. In: User Modelling and User Adapted Interaction, v.6, n. 2-3, p. 87-129, 1996. Special Issue on Adaptive Hypertext and Hypermedia.

Carvalho, A. C. P. L. F.; Faceli, K.; Lorena, A. C.; Gama, J. Inteligência Artificial:- Uma Abordagem de Aprendizado de Máquina. Rio de Janeiro: LTC, 2011.

Charniak, E.; McDermott, D. A Bayesian Model of Plan Recognition. Massachusetts: Addison-Wesley, 1985.

Cheffer, F.; Silveira, S. R. Sistema de Recomendação de Produtos de Pet Shop integrado a uma Loja Virtual. Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica, v. 13, p. 7, 2023. Disponível em: <http://periodicos.unifacel.com.br/resiget/article/view/2672>. Acesso em: 30 abr. 2025.

Coelho, M. M. Definição de um módulo hipermídia adaptativo para integrar Sistemas Tutores Inteligentes. 1995. Disponível em: <http://www.ita.cta.br/~coelho/paper.html>. Acesso em: 15 abr. 2016.

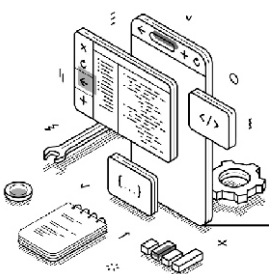
Colli, E.; Silveira, S. R. Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom. Educere et Educare, v. 18, p. 47-67, 2023. Disponível em: <https://e-revista.unioeste.br/index.php/educereeteducare/article/view/30523>. Acesso em: 30 abr. 2025.

Costa, S. W.; Silva, M. C. S. Aquisição de Conhecimento: o grande desafio na concepção de sistemas especialistas. UFRN: 2005. Acesso em: 19 jun. 2017.

Damásio, A. O Erro de Descartes: emoção, razão e cérebro humano. São Paulo: Companhia das Letras, 2012.

De Bra, P. Pros and Cons of Adaptive Hypermedia in Web-based Education. Journal on Cyber Psychology and Behavior, [S.l.], v. 3, n. 1, 2000. Disponível em: <http://online.liebertpub.com/doi/abs/10.1089/109493100316247>. Acesso em: 12 jun. 2016.

E B I T . W e b s h o p p e r s . D i s p o n í v e l e m : http://www.medsobral.ufc.br/pdf/Webshoppers_39.pdf. Acesso em: 08 out. 2022.



Espinoza, F., Höök, K. A WWW Interface to an Adaptive Hypermedia System. 1996. Disponível em: http://www.sics.se/~kia/PAAM_submission.html. Acesso em: 12 abr. 2016.

Falkembach, G. A. M.; Tarouco, L. M. R. Hipermídia Adaptativa: um recurso para a adequação de ambientes e aprendizagem ao perfil do aprendiz. Canoas: ULBRA. Revista Acta Scientiae, v.2, n.1/2, p. 67-75, jan/dez, 2000.

Fausett, L. Fundamentals of Neural Networks: architectures, algorithms and applications. Prentice Hall International, 1994.

Fava, R. Trabalho, Educação e Inteligência Artificial: a era do indivíduo versátil. Porto Alegre: Penso, 2018.

Fernandes, A. M. R. Inteligência Artificial: noções gerais. Florianópolis: Visual Books, 2005.

Ferrari, R. A.; Silveira, S. R.; Freitas, E. P. Ferramenta para Apoio ao Suporte Técnico em uma Rede FTTx empregando Raciocínio Baseado em Casos. Revista de Sistemas e Computação RSC, v. 9, p. 8, 2019. Disponível em: <https://revistas.unifacs.br/index.php/rsc/article/view/5905>. Acesso em: 30 abr. 2025.

Freitas, R. R.; Silveira, S. R.; Bertolini, C. Sistema de Recomendação de Objetos de Aprendizagem baseado em Estilos Cognitivos e na Taxonomia de Bloom. Anais do IX CINAHPA Congresso Internacional de Ambientes Hipermídia para Aprendizagem, 2019. Disponível em: https://files.cercomp.ufg.br/weby/up/688/o/IX_Congresso_Internacional__Ambientes_Hipermi%CC%81dia_Aprendizagem.pdf. Acesso em: 30 abr. 2025.

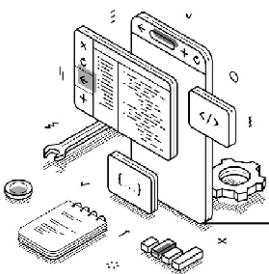
Frizon, M.; Silveira, S. R.; Cunha, G. B. Desenvolvimento de um Protótipo de Sistema Inteligente de Apoio à Decisão como Ferramenta de Business Intelligence: um estudo de caso na Arbaza Alimentos. Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica, v. 6, p. 2, 2015. Disponível em: <http://periodicos.unifacel.com.br/resiget/article/view/1056>. Acesso em: 30 abr. 2025.

Garbin, M.; Silveira, S. R. Sistema de Recomendação Mobile: Um Estudo de Caso para Delivery. Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica, v. 8, p. 4, 2017. Disponível em: <http://periodicos.unifacel.com.br/resiget/article/view/1463>. Acesso em: 30 abr. 2025.

Gasparini, I. Interface Adaptativa no ambiente AdaptWeb navegação e apresentação adaptativa baseada em modelo de usuário. 2003. Disponível em: <http://www.lume.ufrgs.br/handle/10183/12076>. Monografia Acesso em: 22 jun. 2016.

Geller, M. Educação a Distância e Estilos Cognitivos: construindo um novo olhar sobre os ambientes virtuais. Porto Alegre: PGIE/UFRGS, 2004. Tese de Doutorado. Disponível em: http://www.leffa.pro.br/tela4/Textos/Textos/Teses/Marlise_Geller.PDF. Acesso em: 01 mai. 2022.

Guerra, P. A. C.; Silveira, S. R.; Parreira, F.; Ulbricht, V. R. Aplicativo Mobile para avaliar a acessibilidade de Objetos de Aprendizagem utilizando um Sistema Especialista. Revista Educação Especial, v. 33, p. 11, 2020. Disponível em: <https://periodicos.ufsm.br/educacaoespecial/article/view/42711>. Acesso em: 30 abr. 2025.



Haykin, S. Redes Neurais: princípios e prática. Porto Alegre: Bookman, 2007.

IPB - Instituto Pet Brasil. Censo Pet IPB: com alta recorde de 6% em um ano, gatos lideram crescimento de animais de estimação no Brasil. Disponível em: <http://institutopetbrasil.com/fique-por-dentro/amor-pelos-animais-impulsiona-os-negocios-2-2/#:~:text=A%20pesquisa%20revela%20que%20o,3%20milh%C3%B5es%20do%20ano%20anterior>. Acesso em: 07 out. 2022.

Junquilha, T. A. Inteligência Artificial no Direito: limites éticos. Salvador, BA: Juspodivm, 2022.

Kayaer, K.; Yildirim T. Medical diagnosis on Pima Indian diabetes using general regression neural networks. In: Proceedings of the International Conference on Artificial Neural Networks and Neural Information Processing (ICANN/ICONIP), p. 181–184, 2003.

Koch, N. P. Software Engineering for Adaptive Hypermedia Systems – Reference Model, Modeling Techniques and Development Process. PhD Thesis. Ludwig-Maximilians-Universität München. 2001. Disponível em: <http://www.pst.informatik.uni-muenchen.de/~kochn/>. Acesso em: 12 abr. 2016.

Lengert, P. S.; Silveira, S. R. Estudo e Aplicação de Técnicas de Mineração de Dados em uma Base de Dados Não Estruturada sobre o Clima da Região Oeste de SC. UFSM: Frederico Westphalen, 2024.

Lévy, P. A Máquina Universo: criação, cognição e cultura informática. Porto Alegre: Artmed, 1998.

Lorenzi, F.; Silveira, S. R. Desenvolvimento de Sistemas de Informação Inteligentes. Porto Alegre: Ed. UniRitter, 2011.

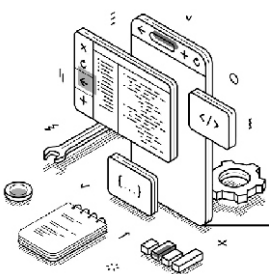
Lourenço. B. M. P. Sistema Especialista Para Auxílio No Diagnóstico De Diabetes Mellitus. Publicações de Monografia no portal da Universidade Presidente Antônio Carlos, Barbacena – MG, 2003. Disponível em: <https://ri.unipac.br/repositorio/wp-content/uploads/tainacan-items/282/195870/PAULA-MOREIRA-BARBOSA-LOURENCO-SISTEMA-ESPECIALISTA-PARA-AUXILIO-NO-DIAGNOSTICO-COMPUTACAO-2003.pdf>. Acesso em: 5 abr. 2017.

Luger, G. F. Inteligência Artificial: Estruturas e estratégias para a solução de problemas complexos. 6. ed. São Paulo: Pearson Universidades, 2013.

Macedo, C. M. S. Diretrizes para Criação de Objetos de Aprendizagem Acessíveis. Florianópolis/SC: PPGEGC/UFSC, 2010. Tese de Doutorado. Disponível em: <https://repositorio.ufsc.br/handle/123456789/94396>. Acesso em 28 jun. 2019.

Massad, E; Meneses, R. X.; Silveira, P. S. S; Ortega, N. R. S. Métodos Quantitativos em Medicina. Barueri, SP: Manole, 2004.

Matanza, J. M. B. Ardora. 2014. Disponível em: <http://webardora.net>. Acesso em: 15 fev. 2014.



McCarthy, J.; Hayes P. J. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In MELTZER, B. e MICHIE D. (eds.) Machine Intelligence, 4. Edinburgo: University Press, 1969.

MERLOT.ORG. MERLOT: Multimedia Educational Resource for Learning and Online Teaching. Disponível em: <https://www.merlot.org/merlot/index.htm>. Acesso em: 27 nov. 2019.

Meyer, J. O Guia Essencial do HTML 5: usando jogos para aprender HTML 5 e JavaScript. Rio de Janeiro: Ciência Moderna, 2011.

Ministério do Meio Ambiente. Caderno de Licenciamento Ambiental. 2009. Disponível em: http://www.mma.gov.br/estruturas/sqa_pnla/_arquivos/ultimo_caderno_pnc_licenciamento_caderno_de_licenciamento_ambiental_46.pdf. Acesso em: 18 abr. 2014.

Motta, C.; Bicharra, A. C.; Vivacqua, A.; Santoro, F. M.; Oliveira, J. Sistemas de Recomendação. Disponível em: https://www.researchgate.net/profile/Jonice-Oliveira/publication/328228374_Sistemas_de_Recomendacao/links/5bbf9ec5a6fdcc2c91f6ac1a/Sistemas-de-Recomendacao.pdf. Acesso em: 30 abr. 2022.

NADTA. Núcleo de Acessibilidade Digital e Tecnologias Assistivas: grupo de pesquisa. Disponível em: <http://dgp.cnpq.br/dgp/espelhogrupo/58948>. Acesso em: 29 jun. 2019.

OPENAI.COM. ChatGPT: Optimizing Language Models for Dialogue. Disponível em: <https://openai.com/blog/chatgpt/>. Acesso em: 18 fev. 2023.

Palazzo, L. A. M. Modelos Proativos para Hipermedia Adaptativa. 2000. Tese de Doutorado em Ciência da Computação. Porto Alegre: PPGC-UFRGS.

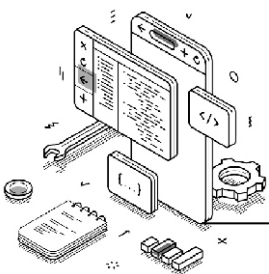
Palazzo, L. A. M. Sistemas de Hipermedia Adaptativa. In: Anais do Congresso da Sociedade Brasileira de Computação, 2002. Florianópolis. Convergências Tecnológicas: redesenhando as fronteiras da ciência e educação Florianópolis: SBC, p.1-38.

Palma Neto, L. G.; Nicoletti, M. C. Introdução às Redes Neurais Construtivas. São Carlos, SP: Editora da Universidade Federal de São Carlos, 2005.

Parreira, F. J.; Falkembach, G. A. M.; Silveira, S. R. Construção de Jogos Educacionais Digitais e Objetos de Aprendizagem: um estudo de caso empregando Flash, HTML 5, CSS, Javascript e Ardora. Rio de Janeiro: Ciência Moderna, 2018.

PASINI, C. G. D.; CARVALHO, E.; ALMEIDA, L. H. C. A Educação Híbrida em Tempos de Pandemia: algumas considerações. OSE – Observatório SocioEconômico da COVID-19. 2020. Disponível em: https://www.osecovid19.cloud.ufsm.br/media/documents/2021/03/29/Textos_para_Discussao_09_-_Educacao_Hibrida_em_Tempos_de_Pandemia.pdf. Acesso em: 30 abr. 2025.

Pereira, A. S.; Bertagnolli, S. C.; Silveira, S. R. Metodologia da Aprendizagem em EaD. Santa Maria: UAB/NTE/UFSM, 2017. Disponível em: <https://repositorio.ufsm.br/handle/1/15809>. Acesso em: 29 jun. 2019.



Peruzzo, D. M.; Silveira, S. R.; Parreira, F. J.; Stock, T. P. Sistema Especialista para auxiliar no Diagnóstico de Depressão. Revista Saúde.com, v. 15, p. 1, 2019.

Prado, K. S.; Silveira, S. R. Framework Genérico de Recomendação para Lojas Virtuais. Revista de Ciência e Tecnologia, v. 1, p. 7, 2015. Disponível em: <https://revista.ufr.br/rct/article/view/2524>. Acesso em: 30 abr. 2025.

Pujol, L. O que o ChatGPT significa para a Educação. 2023. Disponível em: <https://desafiosdaeducacao.com.br/chatgpt-educacao/>. Acesso em: 18 fev. 2023.

Reis, L. F. M. Sistema de Recomendação Baseado em Conhecimento. Disponível em: <https://estudogeral.sib.uc.pt/handle/10316/99769>. Acesso em: 15 out. 2022.

Rezende, S. Sistemas Inteligentes: fundamentos e aplicações. São Paulo: Manole, 2005.

Ribeiro, V. G.; Zabadal, J. R. S. Pesquisa em Computação: uma abordagem metodológica para trabalhos de conclusão de curso e projetos de iniciação científica. Porto Alegre: Editora UniRitter, 2010.

Rich, E.; Knigh, K. Inteligência Artificial. 2. ed. Rio de Janeiro: McGraw-Hill, 1994.

Russell, S.; Norvig, P. Inteligência Artificial. Rio de Janeiro: LTC, 2021.

SAE Digital. Taxonomia de Bloom. Disponível em: <https://sae.digital/taxonomia-de-bloom/>. Acesso em: 01 mai. 2022.

Silva, M. S. Web Design Responsivo. São Paulo: Novatec Editora Ltda, 2014.

Silva, I. N.; Spatti, D. H.; Flauzino, R. A. Redes Neurais Artificiais para Engenharia e Ciências Aplicadas. São Paulo: Artliber, 2010.

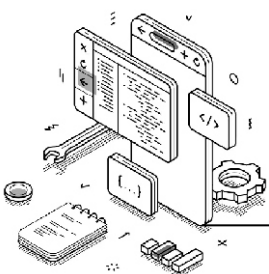
Silveira, S. R.; Vit, A. R. D. Inteligência Artificial: os computadores podem ser criativos? In: Inteligência Artificial e Propriedade Intelectual. Fernandes, M. S.; Caldeira, C. M. G. (Org). Rio de Janeiro: GZ, 2023.

Silveira, S. R.; Parreira, F. J.; Bigolin, N. M. Metodologia do Ensino e da Aprendizagem em Informática. Santa Maria: UAB/NTE/UFSM, 2019. Disponível em: <https://repositorio.ufsm.br/handle/1/18334>. Acesso em: 30 abr. 2025.

Silveira, S. R.; Parreira, F. J.; Souza, N. G.; Souza, A. S. S. Adaptação de Cursos a Distância Empregando Objetos de Aprendizagem. Revista de Empreendedorismo, Inovação e Tecnologia, v. 3, p. 12-21, 2016. Disponível em: <https://seer.atitus.edu.br/index.php/revistasi/article/view/1366/923>. Acesso em: 30 abr. 2025.

Silveira, S. R. Formação de Grupos Colaborativos em um Ambiente Multiagente de Aprendizagem na Internet: um estudo de caso utilizando sistemas multiagentes e algoritmos genéticos. Porto Alegre: PPGC/UFRGS. Tese de Doutorado, 2006. Disponível em: <https://lume.ufrgs.br/handle/10183/7130>. Acesso em: 30 abr. 2025.

Souza, A. S.; Parreira, F. J.; Falkembach, G. A. M.; Souza, N. G.; Silveira, S. R. Alternativas para Construção de Objetos de Aprendizagem: um estudo de caso voltado à definição de uma arquitetura para adaptação de cursos na modalidade de Educação a Distância. Revista de Ciência e Tecnologia, v. 1, p. 1, 2015. Disponível em: <https://revista.ufr.br/rct/article/view/2498>. Acesso em: 30 abr. 2025.



Sujatha, V. An Intelligent Expert Based System Neural Network for the Diagnosis of Type2 Diabetes Patients. School of Advanced Sciences VIT University, India, 2014. Disponível em: <http://ijirae.com/images/downloads/vol1issue2/ACS10106.April14.17.pdf>. Acesso em: 15 jun. 2015.

Teixeira, J. F. Mentis e Máquinas: uma introdução à ciência cognitiva. Porto Alegre: Artes Médicas, 1998.

Thé, M. A. L. Raciocínio Baseado em Casos: uma abordagem fuzzy para diagnóstico nutricional. Tese (Doutorado) - Curso de Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2001. Disponível em: <https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/79411/179202.pdf?sequence=1&isAllowed=y>. Acesso em: 23 maio 2018.

Tissot, H. C.; Camargo, L. C.; Pozo, A. T. R. Treinamento de Redes Neurais Feedforward: comparativo dos algoritmos Backpropagation e Differential Evolution. IX Encontro Nacional de Inteligência Artificial (ENIA-2012). Curitiba: Universidade Federal do Paraná (UFPR). Disponível em: http://www.ppgia.pucpr.br/~enia/anais/enia/artigos/105243_2.pdf. Acesso em: 22 mar. 2015.

Torres, R. Personalização na Internet: como descobrir os hábitos de consumo de seus clientes, fidelizá-los e aumentar o lucro de seu negócio. São Paulo: Novatec, 2004.

UCI. University of California. Machine Learning and Intelligent System, School of Information and Computer Science, 2010. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes>. Acesso em: 17 ago. 2014.

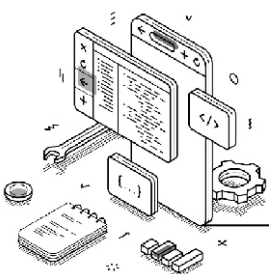
UFSM. Universidade Federal de Santa Maria. Manancial: Repositório digital da UFSM. Disponível em: <https://repositorio.ufsm.br>. Acesso em: 29 jun. 2019.

Vellasco, M. M. B. R. Redes Neurais Artificiais. Laboratório de Inteligência Computacional Aplicada (ICA). Rio de Janeiro: Pontifícia Universidade Católica do Rio de Janeiro (PUC-RJ), 2007. Disponível em: <http://www2.ica.ele.puc-rio.br/Downloads/33/ICAintrodu%C3%A7%C3%A3o%20RNs.pdf>. Acesso em: 22 mar. 2015.

Vieira, J. P.; Parreira, F. J.; Silveira, S. R. Desenvolvimento de um Sistema para Auxílio ao Diagnóstico de Diabetes empregando Redes Neurais (SADD). Revista de Sistemas e Computação RSC, v. 6, p. 3, 2016. Disponível em: <https://revistas.unifacs.br/index.php/rsc/article/view/4148>. Acesso em: 30 abr. 2025.

Vilete, A. S.; Lopes, T. M. Frameworks para o Desenvolvimento de Aplicações Mobile Multiplataforma. 2018. Disponível em: <http://bd.centro.iff.edu.br/jspui/handle/123456789/2184>. Acesso em: 04 mai. 2019.

Vitorino, T. A. S. Raciocínio Baseado em Casos: conceitos e aplicações. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, 2009. Disponível em: https://repositorio.ufmg.br/bitstream/1843/BUOS-8CZLMW/1/thiago_arreguy_silva_vitorino_1.pdf. Acesso em: 30 abr. 2025.



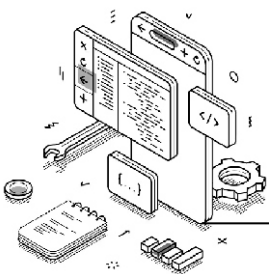
W3C. Web Content Accessibility Guidelines (WCAG) 2.0. 2014. Tradução Autorizada em Português do Brasil. Disponível em: <https://www.w3.org/Translations/WCAG20-pt-br/>. Acesso em: 04 mai. 2019.

Wangenheim, C. G.; Wangenheim, A. Raciocínio Baseado em Casos. Barueri: Manole, 2003.

Wu, H. A Reference Architecture for Adaptive Hypermedia Systems. In: Workshop on Adaptive Hypertext and Hypermedia. 2001 Proceedings.... Disponível em: <http://wwwis.win.tue.nl/ah2001/proceedings.html>. Acesso em: 12 jun. 2016.

Yin, R. K. Estudo de caso: planejamento e métodos. 4.ed. Porto Alegre: Bookman, 2015.

Zuben. J. F. Sistemas Baseados em Regras e Árvores de Decisão. DCA/FEEC/Unicamp, 2011. Disponível em: https://www.dca.fee.unicamp.br/~vonzuben/courses/ea072_2s20/topico6_EA072_2s2020_Part1e2.pdf. Acesso em: 09 abr. 2017.



APRESENTAÇÃO DOS AUTORES



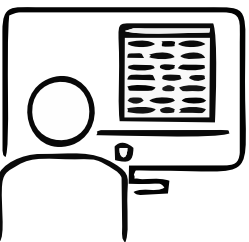
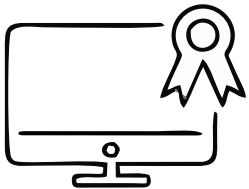
Experiências de Desenvolvimento de Sistemas de Informação Inteligentes

APRESENTAÇÃO DOS AUTORES



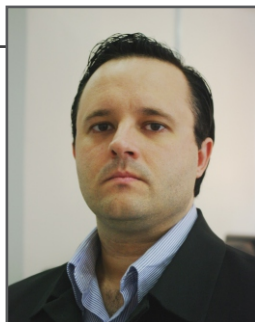
SIDNEI RENATO SILVEIRA

é Professor Associado II do Departamento de Tecnologia da Informação da UFSM (Universidade Federal de Santa Maria) - Campus Frederico Westphalen/RS. Técnico em Processamento de Dados pelo Colégio ULBRA Cristo Redentor. Bacharel em Informática pela ULBRA (Universidade Luterana do Brasil). Especialista em Administração e Planejamento para Docentes pela ULBRA. Especialista em Gestão Educacional pelo SENAC. Mestre em Ciência da Computação pela UFRGS (Universidade Federal do Rio Grande do Sul) na área de Jogos Educacionais Digitais e Sistemas Multimídia. Doutor em Ciência da Computação pela UFRGS na área de Inteligência Artificial. Atualmente está cursando a Especialização em Espaços e Possibilidades para Educação Continuada pelo IFSul (Instituto Federal Sul-Rio-Grandense).



ANTÔNIO RODRIGO DELEPIANE DE VIT

Doutor em Ciência da Computação pela PUCRS, Mestre em Ciência da Computação pela UFRGS e Bacharel em Informática pela Unijuí. Professor Adjunto em regime de Dedicação Exclusiva junto à UFSM/FWJ. Desenvolve pesquisas na área de Ciência da Computação, com ênfase em Segurança da Informação, Redes de Computadores e Sistemas Distribuídos, atuando principalmente nos seguintes temas: Programação de Computadores (C, C++ e Java), Computação de Nós Móveis, Redes Ad Hoc Móveis (MANETs) e Delay or Disruption Tolerant Networks (DTNs). Atua também em Aprendizado de Máquinas (Machine Learning) com ênfase em classificação e previsão de encontros de nós móveis para ambientes de MANETs e DTNs; e classificação e sumarização de documentos textuais não-estruturados. Possui ainda publicações em Informática na Educação em conjunto com o professor Dr. Sidnei Renato Silveira, junto à UFSM/FWJ.



APRESENTAÇÃO DOS AUTORES



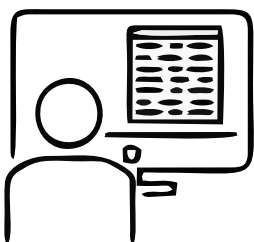
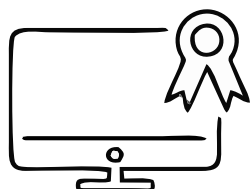
GUILHERME BERNARDINO DA CUNHA

Professor associado III do Departamento de Tecnologia da Informação da UFSM (Universidade Federal de Santa Maria) - Campus Frederico Westphalen/RS. Formado em Ciência da Computação, com especialização em produção de material EAD pela UFAM, com mestrado e doutorado em Ciências com ênfase em Processamento da Informação pela UFU (Universidade Federal de Uberlândia).



NARA MARTINI BIGOLIN

Cientista da Computação, Filósofa, Membro da Ordem Nacional do Mérito Educativo e Professora Associada em Computação na UFSM. Possui graduação em Informática (PUC/RS) e Graduação em Filosofia (Universidade Paulista), Mestrado em Ciência da Computação (UFRGS) e Doutorado em Inteligência Artificial (Sorbonne Université - Pierre et Marie Curie). Atua como Docente e Pesquisadora há 30 anos em várias Universidades da França e do Brasil nas áreas de Ensino da Computação na Educação Básica, em Inteligência Artificial e na temática sobre a desigualdade de gênero nas Olimpíadas Científicas e nos espaços de poder. É Coordenadora do Movimento Meninas Olímpicas do Brasil, coordenadora geral do Torneio Feminino de Computação, líder de equipe da EGDI - Olimpíada Europeia de Informática para Garotas, membro da Comissão da Educação Básica, ex-conselheira suplente da Sociedade Brasileira de Computação e eleita Mulher Cidadã RS na modalidade Educação.



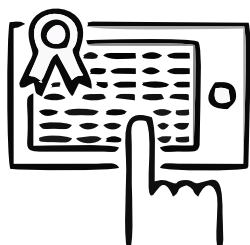
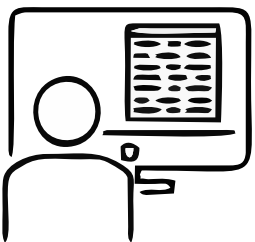
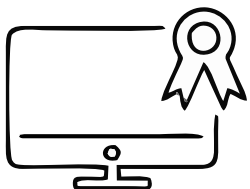
APRESENTAÇÃO DOS AUTORES



VINICIUS GADIS RIBEIRO



Graduado em Ciências Náuticas pelo Ministério da Marinha (1984), e em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (1994), Mestre em Administração pela Universidade Federal do Rio Grande do Sul (1997) e Doutor em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (2005). Atualmente, é professor adjunto do Departamento Interdisciplinar do Campus Litoral Norte da Universidade Federal do Rio Grande do Sul e pertence ao quadro permanente do PPGDesign da mesma universidade. Atuou no Mestrado em Design do Centro Universitário Ritter dos Reis, na área de concentração Design, Educação e Tecnologia. Coordenou o curso de Especialização em Tecnologias Aplicadas a Sistemas de Informação com Métodos Ágeis do UniRitter (2009-2011). Coordenou o Programa de Pós-graduação em Design do Centro Universitário Ritter dos Reis (2011-2015), tendo sido coordenador adjunto em 2017 e 2018. Foi Coordenador Institucional do Programa Ciência sem Fronteiras no UniRitter (2011-2012). É consultor ad-hoc da Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS), na área de Ciência da Computação, e na CAPES, área 29. Membro da comissão de avaliação de periódicos da CAPES, área 29, de 2015 a 2018. Membro da Comissão de Avaliação Quatrienal, área 29, período 2016-2020. Coordenador Adjunto do Programa de Pós-Graduação em Design da UFRGS (2022-2023). Atua nas áreas de Projetos de Segurança Computacional, Métodos de Pesquisa, emprego de Modelos Matemáticos para Otimização de Projetos, emprego de Modelos Matemáticos para Simulações de Fenômenos, Educação em Design e Impactos da Cultura em Design. É autor do livro Fenômenos de Transporte: Fundamentos e Métodos, da Editora Cengage Learning; Pesquisa em Computação: uma abordagem metodológica para Trabalhos de Conclusão de Curso e Projetos de Iniciação Científica, do livro Equações Diferenciais para Engenheiros: uma abordagem prática e do livro Uma introdução à Criptografia: considerações de segurança da informação para o projeto de sistemas com requisitos de privacidade -, da editora UniRitter.



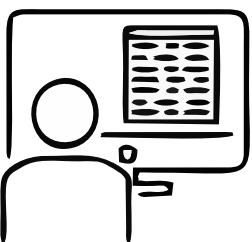
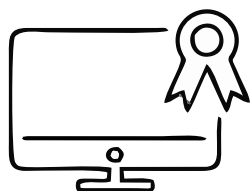
APRESENTAÇÃO DOS AUTORES



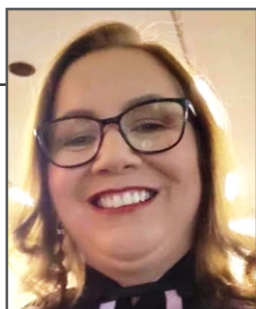
FÁBIO JOSÉ PARREIRA



Professor Titular do Departamento de Tecnologia da Informação da UFSM (Universidade Federal de Santa Maria) - Campus Frederico Westphalen/RS. Possui Graduação em Ciência da Computação pela UNITRI (1999). Mestrado em Processamento Digital de Imagens pela Universidade Federal de Uberlândia (2002). Doutorado em Inteligência Artificial e Informática de Sinais Biomédicos pela Universidade Federal de Uberlândia (2006). Pós-Doutorado (2020) no Laboratório de Mídias e Inclusão Social (LAMID) da UFSC (Universidade Federal de Santa Catarina), atuando em acessibilidade digital e Inteligência Artificial, pela UFSC. Atua na área de Ciência da Computação com enfoque em pesquisas interdisciplinares em Inteligência Artificial, Informática aplicada à saúde e Tecnologia Educacional. Desenvolve pesquisas em Acessibilidade Digital, Jogos Educacionais Digitais e Reconhecimento de Padrões. Tais pesquisas são desenvolvidas nos grupos de pesquisa DIRT-UFSC (Desenvolvimento e Inovação em Reabilitação e Tecnologia Assistiva Digital) e IATE-UFSC (Inteligência Artificial & Tecnologia Educacional).



ADRIANA SADOWSKI DE SOUZA



Graduação em Sistemas de Informação pelo UniRitter - Centro Universitário Ritter dos Reis (2007). Especialização em Tecnologias Aplicadas a Sistemas de Informação pelo UniRitter (2010). Graduação em Licenciatura em Matemática pelo IFFar - Instituto Federal Farroupilha de Frederico Westphalen/RS (2025). Especialização em Metodologia do Ensino de Matemática pelo Centro universitário Unifatecie (2024). Mestranda em Modelagem Matemática e Computacional pela Universidade Regional do Noroeste do Estado do Rio Grande do Sul (UNIJUÍ). Atuou como professora voluntária da 22 edição do Curso de GeoGebra Brasil pela Universidade Estadual do Paraná (Unespar). Atuou como Bolsista do Projeto de Pesquisa e de Monitoramento da Permanência dos estudantes matriculados na Educação de Jovens e Adultos - EJA-EPT (IFFar). Atua como professora substituta do Curso de Ciência da Computação no Instituto Federal Farroupilha (IFFar) - Campus de Frederico Westphalen.



<https://www.facebook.com/Synapse-Editora-111777697257115>



<https://www.instagram.com/synapseeditora>



<https://www.linkedin.com/in/synapse-editora-compartilhando-conhecimento/>



31 98264-1586



editorasynapse@gmail.com



Compartilhando conhecimento